










GUIDE DE PROGRAMMATION


Systemes de Trading (ProBacktest & ProOrder)



SOMMAIRE

 Introduction	6
➔ Accéder à la programmation des systèmes de trading.....	6
➔ La fenêtre de création de systèmes de trading.....	7
> Raccourcis Clavier.....	8
 La Programmation de systèmes de trading	9
➔ La programmation avec ProBuilder.....	9
> Entrées et sorties du marché.....	9
➔ Les Stops et Targets.....	12
> Stops de protection.....	12
> Stops suiveurs.....	14
> Utilisation des "Set Target" / "Set Stop" avec une instruction conditionnelle de type "IF".....	14
> Stops et limites à niveaux multiples.....	15
➔ Arrêter un système de trading avec QUIT.....	17
➔ Suivi des positions.....	17
> Variables d'état.....	17
> Variables de taille de position.....	18
> LongTriggered.....	18
> ShortTriggered.....	18
> TradePrice.....	19
> PositionPerf.....	19
> PositionPrice.....	20
> StrategyProfit.....	20
➔ Définition des paramètres d'exécution des systèmes de trading.....	21
> Cumul des ordres de même direction.....	21
> Précharger de l'historique.....	22
> FlatBefore et FlatAfter.....	23
> Réinvestir les gains (backtests seulement).....	24
> MinOrder et MaxOrder (backtests seulement).....	24
➔ Appeler des d'indicateurs.....	24
> Indicateurs ProRealTime.....	24
> Indicateurs personnalisés.....	24
➔ Astuces de programmation.....	25
> Réduire le nombre d'appels aux indicateurs ProRealTime.....	25
> Appels d'indicateurs personnalisés.....	26
 ProBacktest - La simulation des systèmes de trading	29
➔ Gestion du capital / Money Management.....	30
> Capital initial.....	30
> Frais de courtage et spread.....	30

➔ Optimisation des variables.....	32
➔ Définition de la période d'exécution du backtest.....	34
➔ Personnalisation des plages horaires pour les backtests.....	36
➔ Raisons d'arrêt d'un ProBacktest.....	38
➔ Affichage des valeurs des variables backtest.....	39
 La méthode Walk Forward	41
➔ Cas pratique.....	42
 ProOrder - Le Trading Automatique	46
➔ Préparer un système à l'exécution en trading automatique.....	47
➔ Démarrer un système de trading avec ProOrder et voir sa performance.....	49
➔ Paramètres du Trading Automatique et conditions d'exécution.....	53
> Paramètres des systèmes de trading.....	53
> Arrêt automatique du système de trading.....	54
➔ Interaction entre trading manuel et trading automatique.....	55
➔ Lancer plusieurs systèmes de trading sur un même instrument.....	56
➔ Indicateurs non-utilisables.....	57
➔ Note sur les Fuseaux et heures de marché personnalisés.....	57
 Annexe A : Affichage des résultats	58
➔ Courbe de gains & pertes.....	58
➔ Histogramme des positions.....	59
➔ Rapport détaillé.....	60
 Annexe B : Exemples de codes détaillés	64
> Système de trading basé sur HeikinAshi.....	64
> Système de trading Breakout Range System avec Stop mobile.....	65
> Système de trading basé sur le stochastique lissé.....	66
> Swing Trading, ADX et Moyennes Mobiles.....	67
> Système de trading utilisant un compteur de positions.....	68
> Système de trading utilisant TRADEINDEX – "Inside bar".....	69
➔ Systèmes de trading et Money Management.....	70
> Stops de protection & Objectifs.....	70
> Sortie après inactivité.....	70
> Pyramidage d'une position.....	71
> La martingale classique.....	72
> La grande martingale.....	73
> La Piquemouche.....	74
> La pyramide d'Alembert.....	75
> La contre d'Alembert.....	76

 Annexe C : Exemple détaillé d'un système de trading	77
➔ Présentation du système de trading automatique "Breakout ProOrder".....	78
➔ Résultats du système de trading (historique de 2010 à 2022).....	79
➔ Description des idées du système de trading.....	80
> Idée de départ : le Breakout 30 minutes.....	80
> 2ème idée : deux positions seront prises par jour au maximum.....	81
> 3ème idée : limiter le risque en définissant un écart maximal entre les deux niveaux.....	82
> 4ème idée : augmenter les chances d'exécution favorable.....	83
> 5ème idée : intervenir uniquement sur les breakouts (ou cassures) claires pour éviter les faux signaux.....	84
> 6ème idée : ne pas prendre de position s'il ne reste pas suffisamment de temps.....	85
➔ Code du système de trading "Breakout ProOrder"	86
➔ Comment tester un code / système de trading.....	88
> Mode portefeuille virtuel (PaperTrading).....	88
> Mode trading réel.....	88

 Glossaire	89
--	-----------

Avertissement : ProRealTime n'exerce pas le service de Conseil en Investissement Financier. Ce document n'est en aucun cas une offre de conseil en investissement ni une incitation quelconque à acheter ou vendre des instruments financiers. Les exemples présentés dans ce manuel sont à but pédagogique. Pour votre propre trading, vous êtes entièrement libre dans le choix de vos critères. Les performances passées ne présagent pas des futurs résultats et ne sont pas constantes dans le temps. Tout système de trading peut vous exposer à un risque de perte supérieur à votre investissement initial.

Introduction à la programmation de systèmes de trading sous ProRealTime

L'outil de programmation de ProRealTime permet de créer des stratégies d'investissement personnalisées. Les systèmes de trading peuvent être codés directement ou générés via le module de création assistée (pas de codage requis).

Il existe deux façons d'exécuter vos systèmes :

- En tant que backtests, pour tester leurs performances sur les données historiques d'une valeur
- En tant que système de trading automatique, depuis le module ProOrder : les ordres sont passés en temps réel depuis un compte de trading ou de PaperTrading

La programmation des systèmes de trading utilise le langage de programmation ProBuilder, déjà employé pour la création d'indicateurs personnalisés sous ProRealTime. De nouvelles instructions s'appliquant exclusivement à la création de systèmes de trading ont été ajoutées. La programmation des systèmes de trading inclut des instructions pour les prises de position, les stops et le management du risque de chaque système selon des conditions personnalisées telles que :

- Des indicateurs fournis avec votre plate-forme ou des indicateurs que vous avez programmés vous-même
- Les performances passées de votre système de trading
- Vos derniers ordres

Les résultats de l'exécution d'un système de trading sont présentés sous forme de :

- "Courbe de gains & pertes" (ou "Equity Curve"), qui montre graphiquement l'évolution des gain/pertes de votre système sur l'instrument correspondant.
- Histogramme des positions (vert à l'achat, rouge en vente à découvert). Aucune barre n'est montrée si il n'y a pas de transaction en cours.
- Rapport détaillé, qui vous indique les statistiques générales de votre système pour l'instrument correspondant, sur la période d'exécution choisie.

Dans le cas d'un backtest, il est également possible d'optimiser les variables de votre système. Vous pouvez ainsi sélectionner la combinaison de variables donnant les meilleurs résultats sur l'instrument et la période d'étude.

En trading automatique, les ordres passés par les systèmes de trading apparaissent à la fois dans votre portefeuille PaperTrading et dans la liste des ordres. Les plus-values ou pertes réalisées par vos systèmes sont également reportées dans le portefeuille.

Ce manuel est organisé de la façon suivante : La première partie explique comment créer et éditer un système de trading. La deuxième partie décrit les instructions ProBuilder à utiliser pour leur programmation. La troisième partie détaille la simulation des systèmes avec ProBacktest. Enfin, la quatrième partie présente l'exécution des systèmes en mode trading automatique. Les annexes en fin de manuel donnent plusieurs exemples d'affichage de résultats et de codes, ainsi qu'un glossaire du langage ProBuilder.

Nous conseillons aux utilisateurs débutants ou peu habitués à la programmation de visionner la vidéo intitulée "[Backtester vos stratégies sans écrire une seule ligne de code](#)"

Les informations présentées dans ce manuel sont uniquement destinées à vous aider à coder ou tester vos propres systèmes de trading. Ce ne sont en aucun cas des conseils en investissements.

Pour toutes questions complémentaires sur le fonctionnement du ProBacktest vous pouvez les poser auprès de notre communauté ProRealTime sur le [forum ProRealCode](#), vous y retrouverez aussi une [documentation en ligne](#) avec de nombreux exemples.

En vous souhaitant nos meilleurs vœux de réussite, Bonne lecture !

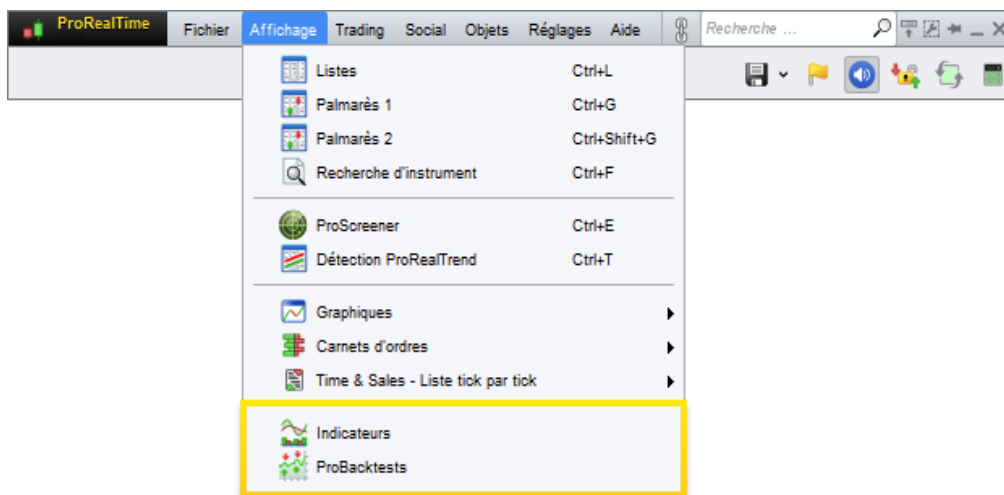
Introduction

Accéder à la programmation des systèmes de trading

La fenêtre de création de systèmes de trading est disponible à partir du bouton "Indicateurs" qui se trouve en haut¹ de chacun de vos graphiques ProRealTime.



Vous pouvez aussi y accéder directement depuis votre menu « Affichage »¹:



Par défaut, la fenêtre de gestion s'ouvre sur l'onglet "Indicateurs". Placez-vous sur le deuxième onglet "Pro-Backtest & Trading Automatique". Vous pourrez alors :

- Accéder à la liste des systèmes existants (prédéfinis ou personnalisés)
- Créer un nouveau système, à appliquer ensuite à la valeur de votre choix
- Modifier, supprimer ou dupliquer un système existant
- Importer ou exporter des systèmes
- Ajouter vos codes achetés sur la [Marketplace ProRealCode](#)

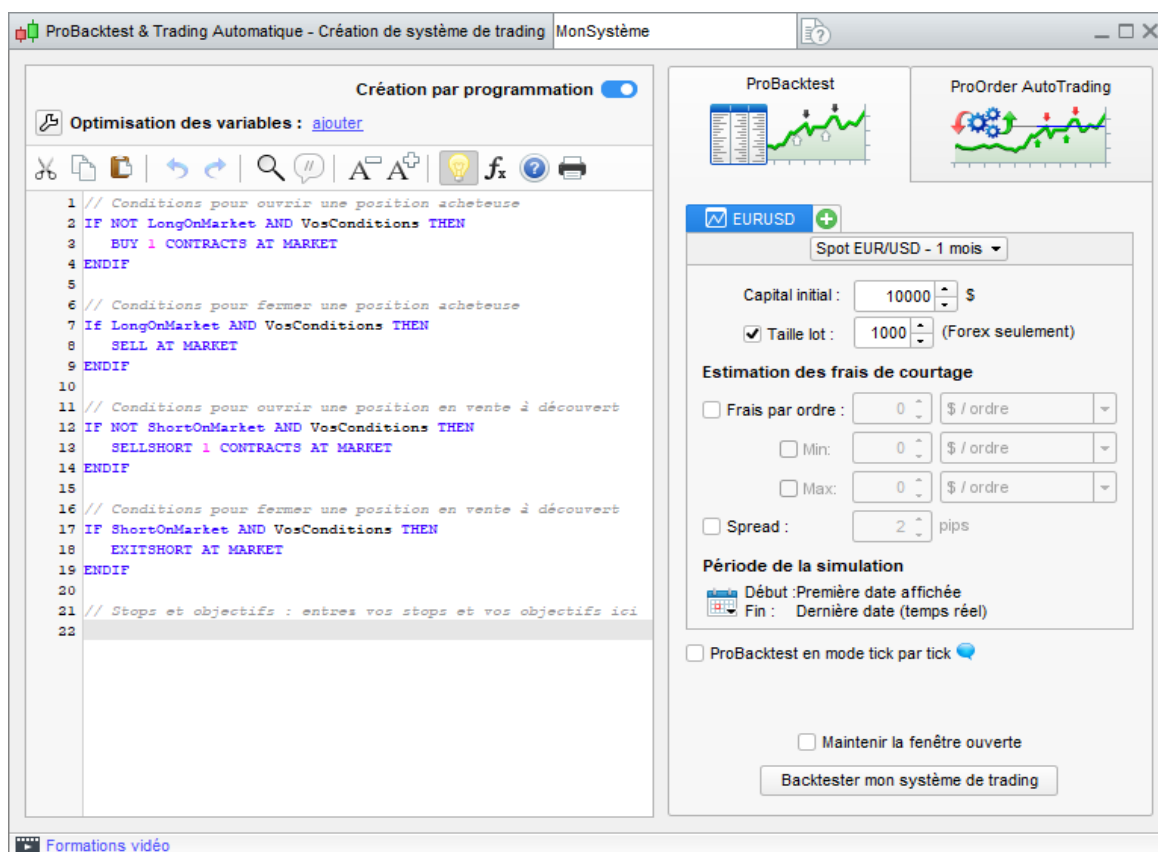
¹ Depuis la version 12 de ProRealTime

La fenêtre de création de systèmes de trading

La fenêtre de création de systèmes de trading est composée de deux zones principales :

- La zone de création (assistée ou par programmation) dans la partie gauche de la fenêtre.
- La zone d'application dans la partie droite, qui comprend l'onglet ProBacktest pour la simulation de systèmes et l'onglet ProOrder pour leur exécution en trading automatique. Le paramétrage du ProBacktest est détaillé dans la section 3 de ce manuel.

La zone de création, vous permet de :

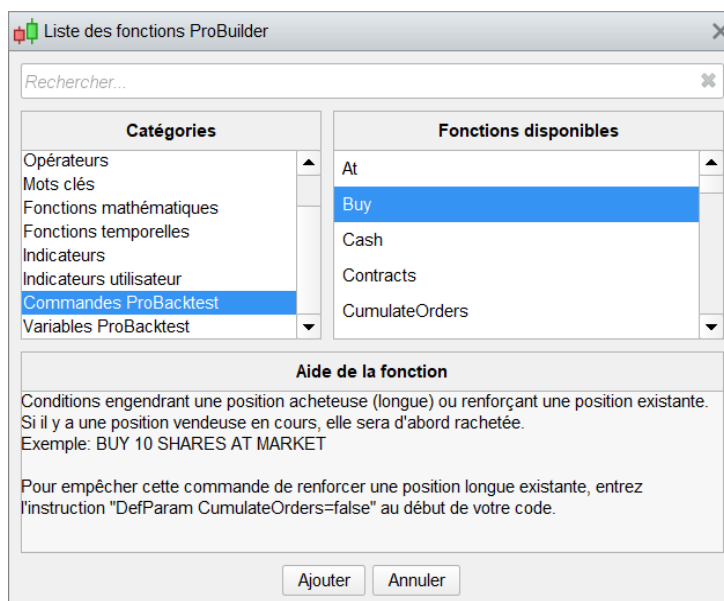


- Programmer directement un système de trading dans la zone de texte.
- Vous servir du bouton "Insérer Fonction", qui ouvre dans une nouvelle fenêtre la bibliothèque des fonctions disponibles, afin de vous aider lors de la programmation. Vous y retrouverez également des explications liées à la commande ou fonction choisie.

Exemple :

Utilisons la bibliothèque de fonctions en cliquant sur "Insérer Fonction".

Placez-vous sur la section "Commandes ProBacktest" et cliquez sur "**BUY**", puis cliquez sur le bouton "Ajouter". La commande s'ajoutera ainsi à la zone de programmation.



Essayons donc de créer une ligne complète de code. Supposons par exemple qu'on souhaite acheter 10 titres au prix du marché.

Procédez de la même façon qu'au-dessus pour récupérer dans cet ordre les instructions "**SHARES**", "**AT**" et "**MARKET**" (vous séparerez chaque mot d'un espace)

Vous obtenez alors l'instruction "**BUY 10 SHARES AT MARKET**" qui commande l'achat de 10 titres au marché. La section suivante présente toutes les instructions disponibles pour la programmation de systèmes de trading.

Des exemples de codes de systèmes complets sont disponibles dans l'annexe B de ce manuel.

Raccourcis Clavier

Dans ProRealTime version 12, la fenêtre de création de systèmes de trading possède plusieurs fonctionnalités pratiques utilisables via des raccourcis clavier :

- Tout sélectionner (Ctrl+A) : Sélectionne tout le texte présent dans l'éditeur de code
- Copier (Ctrl + C) : Copie le texte sélectionné
- Coller (Ctrl + V) : Colle le texte copié
- Annuler (Ctrl + Z) : Annule la dernière action faite dans l'éditeur de code
- Refaire (Ctrl + Y) : Refait la dernière action faite dans l'éditeur de code
- Rechercher / Remplacer (Ctrl + F) : Cherche un texte dans l'éditeur de code / Remplace un texte dans l'éditeur de code
- Commenter / Dé-commenter (Ctrl + R) : Commente le code sélectionné / Dé-commente le code sélectionné (le code commenté sera précédé de « // » ou entouré de « /* */ » et coloré en gris. Il ne sera pas pris en compte lors de l'exécution du code.
- Activer l'aide à la complétion (Ctrl + Espace) : Permet d'obtenir une aide sur les fonctionnalités disponibles en fonction du début du mot que l'utilisateur écrit. Il peut-être activé/désactivé de manière permanente

Pour les utilisateurs de MAC, les mêmes raccourcis claviers peuvent être utilisés. Il suffit dans ce cas de remplacer la touche « Ctrl » par la touche « Command ».

La plupart de ces raccourcis peuvent aussi être utilisés via un clic droit dans l'éditeur de code de la fenêtre de création du système de trading.

La Programmation de systèmes de trading

Avertissement : Les exemples présentés dans ce manuel sont à but pédagogique. Pour votre propre trading, vous êtes entièrement libre dans le choix de vos critères. Toutes les informations présentées sont à caractère "général" et ne sauraient, en aucun cas, représenter de quelconques informations ou conseils personnalisés ou une quelconque incitation à acheter ou vendre des instruments financiers. Les performances passées ne présagent pas des futurs résultats et ne sont pas constantes dans le temps. Tout système de trading peut vous exposer à un risque de perte supérieur à votre investissement initial.

La programmation avec ProBuilder

Il est fortement recommandé de lire le [manuel de programmation ProBuilder](#) consacré à la programmation d'indicateurs. La présente section s'intéresse aux instructions ProBuilder complémentaires applicables aux systèmes de trading.

Pour rappel, ProBuilder est le langage de programmation de ProRealTime. Ce langage est très simple d'utilisation et offre de nombreuses possibilités. Il est basé sur les principes suivants :

- L'unité de base est le chandelier, et l'unité de temps est calquée sur celle du graphique en cours.
- Un programme ProBuilder et toutes ses sous-fonctions sont réévaluées à la clôture de chaque chandelier, depuis le plus ancien vers le plus récent.
- Toutes les instructions d'achat / vente sont calculées à la fin du chandelier en cours. Elles seront donc déclenchées au plus tôt à l'ouverture du chandelier suivant.

Entrées et sorties du marché

Différentes instructions s'appliquent selon le sens de votre position :

Positions longues :

- BUY : instruction d'achat de titres (entrée longue)
- SELL : instruction de vente des titres achetés (sortie longue)

L'instruction BUY permet d'ouvrir (ou de renforcer) une position longue sur le marché. Elle est associée à l'instruction SELL qui permet de fermer (ou d'alléger) une position longue. L'instruction SELL n'a pas d'effet si aucune position longue n'est ouverte.

Positions courtes :

- SELLSHORT : instruction de vente à découvert de titres (entrée courte)
- EXITSHORT : instruction de rachat des titres vendus à découvert (sortie courte)

Ces deux instructions fonctionnent de manière symétrique avec BUY et SELL. L'instruction EXITSHORT n'a pas d'effet si aucune position courte n'est ouverte.

La prise de position long et short simultanée sur une même valeur n'est pas possible. Dans la pratique, il est donc possible de fermer une position longue par une instruction SELLSHORT, ou de fermer une position courte avec une commande BUY.

Remarque :

En trading automatique, pensez à vérifier que la taille de position maximale définie au lancement n'est pas inférieure à la taille de position que votre système de trading essaiera d'ouvrir (ou renforcer) .Dans le cas contraire, l'ordre ne sera pas transmis et donc ne sera pas exécuté,

Chacune de ces commandes peut être complétée des éléments suivants, qui seront détaillés par la suite :

<Ordre> <Quantité> AT <type>

Exemple :

BUY 1000 CASH AT MARKET **ou** SELL 1 SHARE AT 1,56 LIMIT

Quantité

Il existe 2 façons de définir la quantité à investir :

- **SHARES** qui correspond à une unité de l'instrument : "1 share" représente 1 action, 1 warrant, 1 contrat pour les futures, ou 1 lot sur Forex. SHARES peut être remplacé à tout moment par SHARE, CONTRAT, CONTRACTS, LOT ou LOTS sur n'importe quel type d'instruments. Dans le cas du Forex, la quantité demandée sera multipliée par la taille d'un lot. Si la quantité n'est pas renseignée, les valeurs prises par défaut sont :
 - 1 unité pour une entrée en position (Ex : BUY AT MARKET place un ordre de "1" au marché)
 - La quantité totale de la position pour une sortie (Ex : SELL AT MARKET ferme entièrement la position)
- **CASH** correspond à une transaction d'un montant déterminé (en € ou \$ par exemple) uniquement sur les actions. La quantité correspondante de l'ordre sera alors calculée lors de la clôture de la barre (arrondie par défaut) à l'unité inférieure. Les frais de courtage ne sont pas pris en compte lors du calcul de la quantité correspondant au montant à acheter ou à vendre.

Exemple : BUY 1000 CASH AT MARKET

Il est possible d'utiliser l'instruction ROUNDEDUP pour arrondir cette quantité à l'unité supérieure ou ROUNDEDOWN pour arrondir cette quantité à l'unité inférieure.

Exemple : BUY 1000 CASH ROUNDEDUP AT MARKET

Modes

Trois modes d'exécution d'ordres sont à votre disposition :

AT MARKET : L'ordre sera passé au prix du marché à l'ouverture de la barre suivante.

Exemple : BUY 1 SHARE AT MARKET

AT <prix> LIMIT : Un ordre limite sera posé au prix indiqué

AT <prix> STOP : Un ordre stop sera posé au prix indiqué

Exemple : BUY 1 SHARE AT 10.5 LIMIT

Les ordres limites et stops à niveaux de prix déterminés sont valables par défaut pour une durée d'une barre, à partir de l'ouverture de la barre suivante. Ils sont donc annulés à sa clôture s'ils n'ont pas été exécutés.

Ces ordres sont différents des stops de protection et limites profits (voir section suivante). Les stops de protections et les limites profits sont liés à une position déjà ouverte, et valables jusqu'à la clôture de celle-ci .

Ces ordres peuvent être assimilés à des ordres au marché si les conditions le permettent :

- Si un ordre BUY <quantité> AT <prix> LIMIT est placé au dessus du prix actuel, il sera traité comme un ordre au marché.
- Si un ordre BUY <quantité> AT <prix> STOP est placé en dessous du prix actuel, il sera traité comme un ordre au marché.
- Si un ordre SELLSHORT <quantité> AT <prix> LIMIT est placé en dessous du prix actuel, il sera traité comme un ordre au marché.
- Si un ordre SELLSHORT <quantité> AT <prix> STOP est placé au dessus du prix actuel, il sera traité comme un ordre au marché.

Exemple :

Le système suivant prévoit l'achat d'1 action au prix du marché lorsque le RSI est en zone de survente (RSI < 30) et que les cours sont en dessous de la bande de Bollinger inférieure.

Lorsque le RSI est en zone de surachat (RSI > 70) et les cours sont au-dessus de la bande de Bollinger supérieure, alors on vend.

```
MyRSI = RSI[14](Close)
MyBollingerDown = BollingerDown[25](Close)
MyBollingerUp = BollingerUp[25](Close)

IF MyRSI < 30 AND Close < MyBollingerDown THEN
    BUY 1 SHARE AT MARKET
ENDIF
IF MyRSI > 70 AND Close > MyBollingerUp THEN
    SELL AT MARKET
ENDIF
```



Vous pouvez définir la durée de validité de vos ordres stops et limites.

L'exemple suivant illustre l'utilisation de certaines variables pour déterminer la durée de validité d'un ordre limite. La validité s'exprime en nombre de barres. Ici, le code place un ordre limite d'achat au prix de clôture de la barre sur laquelle un croisement de moyenne mobile a lieu. L'ordre limite reste valide 10 barres après le croisement. S'il n'est pas exécuté pendant cette période, il sera annulé.

Exemple :

```
// Définition de la durée de validité de l'ordre
ONCE NbBarLimit = 10

MM20 = Average[20](close)
MM50 = Average[50](close)
// Si la MM20 croise à la hausse la MM50, nous définissons 2 variables : "MyLimitBuy" et
// "MyIndex", qui contiennent respectivement le prix de clôture actuel et l'index de la
// barre concernée par le croisement.
IF MM20 CROSSES OVER MM50 THEN
    MyLimitBuy = close
    MyIndex = Barindex
ENDIF

IF BarIndex >= MyIndex + NbBarLimit THEN
    MyLimitBuy = 0
ENDIF
// Place un ordre au prix correspondant de MyLimitBuy, valable tant que cette variable
// est supérieure à 0 et que nous ne sommes pas en position long.
// Rappel : MyLimitBuy est supérieur à 0 pendant les 10 barres qui suivent le croisement
IF MyLimitBuy > 0 AND NOT LongOnMarket THEN
    BUY 1 SHARES AT MyLimitBuy LIMIT
ENDIF
```

Si un ordre n'est pas exécuté, il est possible de remplacer l'ordre d'achat limite expiré par un ordre d'achat au marché. Le bloc de code vous en donnera la possibilité s'il est ajouté à la suite du bloc précédent :

```
IF MyIndex + NbBarLimit AND MyLimitBuy > 0 AND NOT LongOnMarket THEN
    BUY 1 SHARES AT MARKET
ENDIF
```

Les Stops et Targets

ProBacktest vous permet également de définir des objectifs de profit et des stops de protection. La syntaxe à utiliser est la suivante :

```
SET STOP <type> <valeur> ou SET TARGET <type> <valeur>
```

Exemple :

```
SET STOP %LOSS 2 // Placera un stop loss de 2%
```

Chaque variante est décrite ci-dessous.



Il est important de distinguer les différents ordres **STOP** :

- **AT (prix) STOP**, sert à définir un ordre à seuil d'exécution. Cette ordre est valable pendant une seule barre.
- **SET STOP LOSS (distance au prix) / SET STOP (prix)**, sert à définir un stop de protection, c'est à dire à sortir d'une position existante. Cet ordre est valable jusqu'à la clôture de la position.

Stops de protection

Les stops de protection (stops loss) permettent de limiter les pertes d'une position. Ils peuvent être définis de façon relative ou absolue :

SET STOP LOSS x : Un stop de protection est placé à x unités du prix d'entrée en position.

SET STOP pLOSS x : Un stop de protection est placé à x points du prix d'entrée en position.

SET STOP %LOSS x : Un stop de protection est placé à un niveau correspondant à une perte de x%.

SET STOP \$LOSS x : Un stop de protection est placé à un niveau correspondant à une perte de x€ ou \$ (devise de l'instrument), frais de courtage non compris.

Les stops de protection peuvent aussi être placés au prix d'entrée de la position, permettant de sécuriser une position en empêchant toute perte de celle-ci :

SET STOP BREAKEVEN: Un stop de protection est placé au prix d'entrée en position.

Les stops de protection peuvent enfin être placés de façon à sécuriser un gain lorsque la position est actuellement gagnante. Ils sont alors placés dans le sens du gain de la position. Ils peuvent être définis de façon relative ou absolue :

SET STOP PROFIT x : Un stop de protection est placé à x unités du prix d'entrée en position.

SET STOP pPROFIT x : Un stop de protection est placé à x points du prix d'entrée en position.

SET STOP %PROFIT x : Un stop de protection est placé à un niveau correspondant à un gain de x%.

SET STOP \$PROFIT x : Un stop de protection est placé à un niveau correspondant à un gain de x€ ou \$ (devise de l'instrument), frais de courtage non compris.

Finalement il est possible de définir un stop directement à un prix défini, ce stop pourra donc être placé dans le sens d'une perte ou d'un gain pour la position :

SET STOP PRICE x : Un stop de protection est placé au prix de X sur l'instrument.

La direction (achat/vente) ainsi que la quantité de l'ordre stop sont automatiquement adaptées à la position en cours. Un stop de protection est forcément lié à une position donnée. Si aucune position n'est ouverte, le stop de protection associé ne sera pas actif.

Pour désactiver un stop, il faut utiliser l'une des instructions suivantes :

```
SET STOP LOSS 0, SET STOP [p/%/$]LOSS 0, SET STOP BREAKEVEN 0, SET STOP PROFIT 0, SET STOP [p/%/$]PROFIT 0, SET STOP PRICE 0
```

Attention lors de la pose des stops de protection de position, comme pour les ordres d'ouverture de type STOP et LIMIT vus précédemment, un stop de protection placé du mauvais côté du cours sera transformé automatiquement en ordre au marché de fermeture de position.

Ordres targets (limite)

Ces instructions permettent de sortir de position lorsque les gains atteignent l'objectif déterminé.

SET TARGET PROFIT x : Une limite est placée à x unités du prix d'entrée en position.

SET TARGET pPROFIT x : Une limite est placée à x points du prix d'entrée en position.

SET TARGET %PROFIT x : Une limite est placée à un niveau correspondant à un gain de x%.

SET TARGET \$PROFIT x : Une limite est placée à un niveau correspondant à un gain de x€ ou \$ (devise de l'instrument), frais de courtage non compris.

En cas de position perdante, on peut limiter la perte avec une limite placée au prix d'entrée, qui permettra de sortir de position sans gain ni perte (hors frais de courtage) :

SET TARGET BREAKEVEN : Une limite est placée au prix d'entrée en position.

Il est aussi possible de limiter ses pertes lorsqu'une position est actuellement perdante. La limite est alors placée dans le sens de la perte de la position :

SET TARGET LOSS x : Une limite est placée à x unités du prix d'entrée en position.

SET TARGET pLOSS x : Une limite est placée à x points du prix d'entrée en position.

SET TARGET %LOSS x : Une limite est placée à un niveau correspondant à une perte de x%.

SET TARGET \$LOSS x : Une limite est placée à un niveau correspondant à une perte de x€ ou \$ (devise de l'instrument), frais de courtage non compris.

Enfin comme pour les stops de protection, il est possible de placer une limite de profit directement à un prix défini :

SET TARGET PRICE x : Une limite est placée au prix de X sur l'instrument.

La direction (achat/vente) ainsi que la quantité de l'ordre target sont automatiquement adaptées à la position en cours. Un ordre target est forcément lié à une position donnée. Si aucune position n'est ouverte, l'ordre target associé ne sera pas actif.

Pour désactiver une limite profit dans le code, vous pouvez utiliser l'une des instructions suivantes :

```
SET TARGET PROFIT 0, SET TARGET [p/%/$]PROFIT 0, SET TARGET BREAKEVEN 0, SET TARGET LOSS 0, SET TARGET [p/%/$]LOSS 0, SET TARGET PRICE 0
```

Attention lors de la pose des ordres target, comme pour les ordres d'ouverture de type STOP et LIMIT vus précédemment ainsi que pour les stops de protection de position, un ordre target placé du mauvais côté du cours sera transformé automatiquement en ordre au marché de fermeture de position.

Stops suiveurs

Un stop suiveur (trailing stop) est un ordre stop dont le niveau d'exécution change en fonction de l'évolution du prix. Pour des positions longues, le stop suivra la progression du prix en cas de montée du cours. Si par la suite le cours diminue, le niveau du stop ne changera pas.

Pour les positions courtes le fonctionnement inverse se produit : le stop suivra la baisse des prix, mais restera fixe en cas de remontée du cours.

Sur le même principe que les stops de protection, les stops suiveurs peuvent être définis de façon relative ou absolue :

SET STOP TRAILING y : Un stop suiveur est placé à y unités du prix d'entrée en position.

SET STOP pTRAILING y : Un stop suiveur est placé à y points du prix d'entrée en position.

SET STOP %TRAILING y : Un stop suiveur est placé à un niveau initial correspondant à une perte de y%.

SET STOP \$TRAILING y : Un stop suiveur est placé à un niveau initial correspondant à une perte de y€ ou \$ (devise de l'instrument), frais de courtage non compris.

La direction et la quantité de l'ordre stop suiveur sont automatiquement adaptées à la position en cours. Un stop suiveur est forcément lié à une position donnée. Si aucune position n'est ouverte, le stop suiveur associé ne sera pas actif.

Si la quantité de la position change (on renforce la position par exemple), le seuil du stop est réinitialisé.

Pour désactiver un trailing stop dans le code, vous pouvez utiliser l'une des instructions suivantes :

```
SET STOP TRAILING 0, SET STOP pTRAILING 0, SET STOP %TRAILING 0, SET STOP $TRAILING 0
```

Exemple :

Vous entrez en position sur le future DAX à un niveau de 6000 points et placez un trailing stop à 50 points :

```
SET STOP pTRAILING 50
```

Le stop est initialement placé à 5950. Si le cours monte jusque 6010 puis descend jusque 5980, le stop montera donc de 10 points jusqu'à 5960 et y restera maintenu jusqu'à ce que le cours dépasse les 6010. Il sera donc déclenché si le cours atteint 5960.

Utilisation des "Set Target" / "Set Stop" avec une instruction conditionnelle de type "IF"

Il est possible de modifier le comportement des stops de protection ou limites profit présent dans votre code en y ajoutant des conditions personnalisées. La commande "IF" pourra alors être utilisée.

Exemple :

```
// Place un stop loss de 10% si le gain de la position précédente est d'au moins 10%. Si-  
non, place un stop loss de 5%  
IF PositionPerf(1) > 0.1 THEN  
    SET STOP %LOSS 10  
ELSE  
    SET STOP %LOSS 5  
ENDIF
```

Stops et limites à niveaux multiples

En règle générale, seule une instruction de type "Set Stop" et une instruction de type "Set Target" peuvent être entrées et retenues au sein d'un code. Si votre code comporte plusieurs instructions successives de type "Set Stop" ou "Set Target", seule la dernière commande entrée sera retenue.

Les stops à niveaux multiples sont utilisables en backtest et en trading automatique sur notre module de Pa-perTrading uniquement.

Les stops à niveaux multiples permettent de combiner un stop classique et un stop trailing en même temps sur une position, le plus proche des deux stops sera réellement placé sur le marché, avec la possibilité que le stop trailing prenne automatiquement la place du stop classique selon les mouvements du marché

Exemple :

```
SET STOP %LOSS 10 // Place un stop loss de 10%
SET TARGET PROFIT 50 // Place une limite profit de 50 unités
SET TARGET %PROFIT 5 // Remplace la dernière limite profit par une limite profit de 5%
SET STOP %TRAILING 2 // Remplace le dernier stop loss par un stop suiveur de 2%
```

Il est cependant possible de combiner stops fixes et stops suiveurs, ou stops loss et stops suiveurs. Une instruction unique a été développée dans ce but :

SET STOP	<u><Mode> <valeur></u>	<u><TrailingType> <valeur></u>
	seuil	trailing

Mode : LOSS, pLOSS, %LOSS, \$LOSS

TrailingType : TRAILING, pTRAILING, %TRAILING, \$TRAILING

Cette instruction se présentera donc sous la forme :

```
SET STOP [LOSS/pLOSS/$LOSS/%LOSS] <value> [TRAILING/pTRAILING/$TRAILING/%TRAILING] <value>
```

Exemples d'utilisation:

`SET STOP LOSS x TRAILING y` : Un stop de protection est placé à x **unités** du prix d'entrée en position. Suite à une variation de prix, si l'écart entre le stop suiveur et le prix actuel est inférieur à l'écart entre le stop de protection et le prix actuel, alors le stop de protection devient un stop suiveur de y **unités**.

`SET STOP LOSS x pTRAILING y` : Un stop de protection est placé à x **unités** du prix d'entrée en position. Suite à une variation de prix, si l'écart entre le stop suiveur et le prix actuel est inférieur à l'écart entre le stop de protection et le prix actuel, alors le stop de protection devient un stop suiveur de y **points**.

`SET STOP LOSS x $TRAILING y` : Un stop de protection est placé à x **unités** du prix d'entrée en position. Suite à une variation de prix, si l'écart entre le stop suiveur et le prix actuel est inférieur à l'écart entre le stop de protection et le prix actuel, alors le stop de protection devient un stop suiveur de y **€ ou \$** (devise de l'instrument).

`SET STOP LOSS x %TRAILING y` : Un stop de protection est placé à x **unités** du prix d'entrée en position. Suite à une variation de prix, si l'écart entre le stop suiveur et le prix actuel est inférieur à l'écart entre le stop de protection et le prix actuel, alors le stop de protection devient un stop suiveur de y **%**.

Exemple :

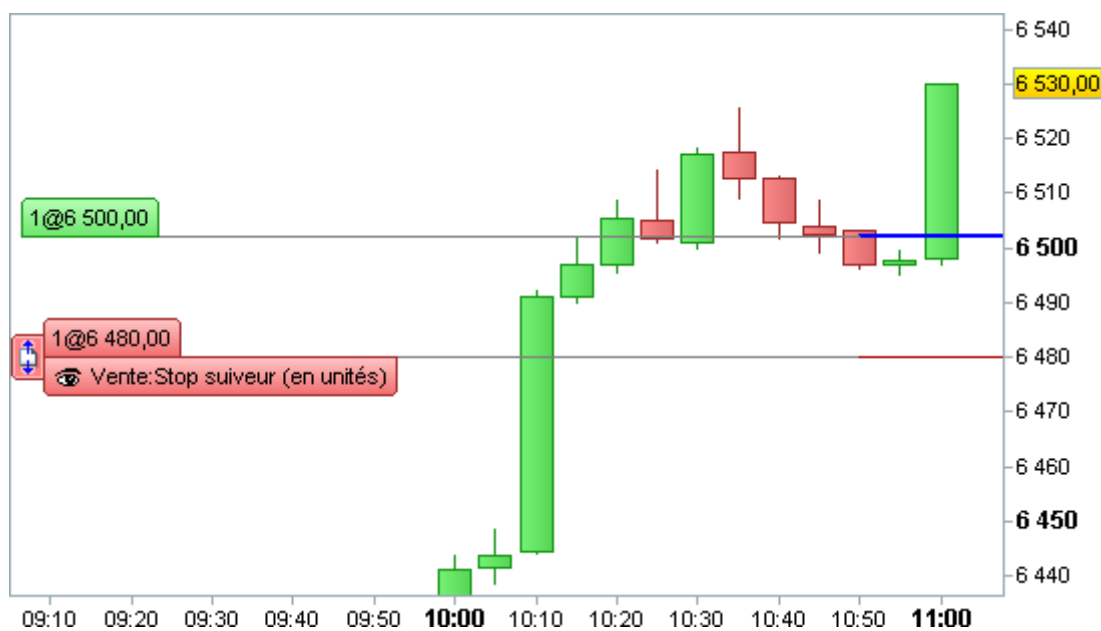
SET STOP LOSS x TRAILING y :

Un stop de protection est placé à un niveau de prix x. Si le niveau du stop suiveur paramétré augmente et dépasse x, alors le stop de protection devient un stop suiveur de y unités.

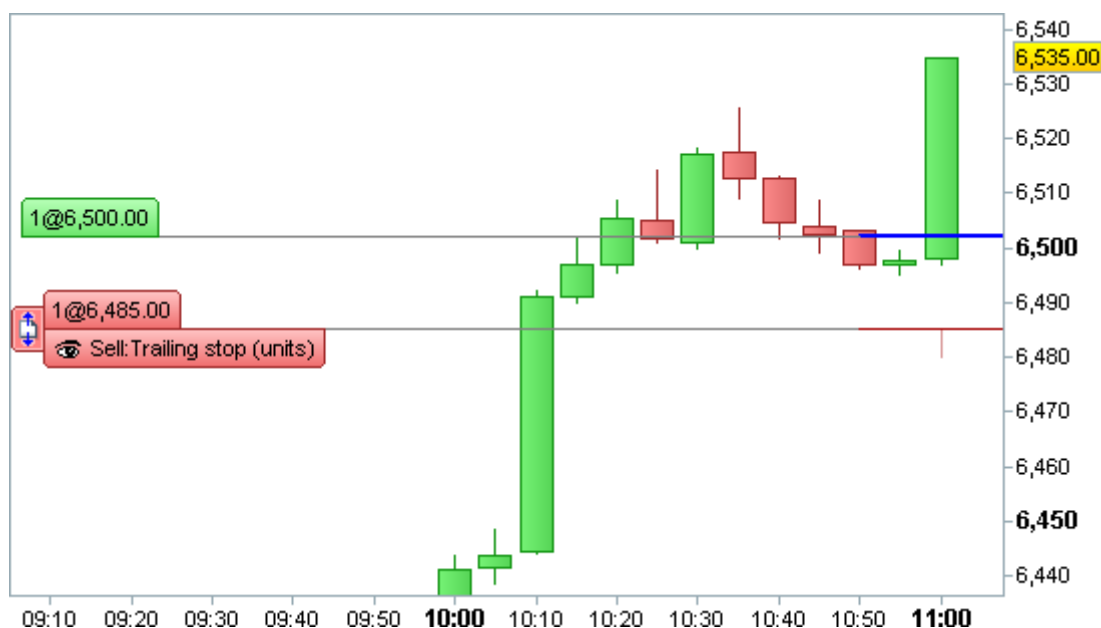
Supposons que vous entriez en position sur le DAX à 6500. Le code suivant placera un stop de protection à 20 points de votre niveau d'entrée en position. Ce stop deviendra un stop suiveur si le prix dépasse les 6530.

```
SET STOP LOSS 20 TRAILING 50
```

Les graphiques suivantes illustrent l'exemple : Le stop est placé à 20 points du cours d'ouverture, soit 6480.



Si le cours atteint 6530 (= 6500 + (50-20)), le stop devient un stop suiveur avec un écart de 50 points.



Si le cours monte (ici ; il est monté à 6535), le stop suiveur monte également : il monte à 6485.

Précision sur l'utilisation des points ou des unités pour les distances de stops et limites

La taille du point varie en fonction des types d'instruments, alors que la taille d'une unité (variation de cours de 1 sur le graphique) est évidemment toujours fixe. En fonction de votre code, et des instruments sur lequel il est appliqué, il peut être préférable d'utiliser des distances en points ou en unités. Exemple :

Sur l'Eurodollar (EURUSD), 1 point = 1 pip = 0,0001

Sur les futures sur indices (DAX, FCE), 1 point = 1

Sur les futures sur taux européens, 1 point = 0,01 (un point de base)

Cette information est disponible dans notre langage grâce au mot-clé POINTSIZE.

Arrêter un système de trading avec QUIT

L'instruction « QUIT » vous permet d'arrêter un système de trading. L'arrêt prend effet à la clôture de la barre en cours. Les ordres en attente sont alors annulés et toutes les positions ouvertes seront clôturées si les paramètres de trading de votre plateforme le demandent. Ceci vous permet d'arrêter facilement un système de trading en cas de larges pertes ou après une certaine date par exemple.

Exemple :

```
If date > 20220101 THEN // Stoppe la stratégie après le 1er Janvier 2022.
    QUIT
ENDIF
```

Suivi des positions

Variables d'état

Les 3 variables permettant de gérer l'état d'un système de trading sont les suivantes :

ONMARKET : vaut 1 si vous avez des positions ouvertes, 0 sinon

LONGONMARKET : vaut 1 si vous avez des positions longues ouvertes, 0 sinon

SHORTONMARKET : vaut 1 si vous avez des positions courtes ouvertes, 0 sinon

Elles peuvent être utilisées avec des crochets. Par exemple, ONMARKET[1] vaut 1 si des positions étaient ouvertes à la clôture de la barre précédente, 0 sinon.

Ces variables sont souvent utilisées avec des instructions de type "IF" avant une entrée en position :

Exemple :

```
// On définit le MACD
Indicator1 = MACD[12,26,9](Close)
// On observe les changements de signe de l'Histogramme du MACD
c1 = (Indicator1 CROSSES OVER 0)
// Achat : Si il n'y a pas de position longue en cours et si MACD >0 , on achète 3 actions ou lots
IF NOT LONGONMARKET AND c1 THEN
    BUY 3 SHARES AT MARKET
ENDIF
```

Variables de taille de position

Ces 3 variables permettent de connaître la quantité de la position en cours :

COUNTOFPOSITION : donne la taille de la position (en lots, titres, ou contrats). Positif si position longue en cours, négatif si position courte en cours.

COUNTOFLONGSHARES : donne la taille de la position (en lots, titres, ou contrats) si une position longue est en cours, 0 sinon.

COUNTOFSHORTSHARES : donne la taille de la position (en lots, titres, ou contrats) si une position courte est en cours, 0 sinon.

Comme pour les instructions vérifiant l'état des positions, ces commandes s'utilisent généralement avec des instructions de type "IF" avant d'entrer en position, et peuvent comporter des crochets.



Rappel au sujet des variables d'état : le code est évalué en fin de chandelier, et les ordres sont passés à l'ouverture de la barre suivante,

Dans le code suivant, la variable "long" ne sera pas égale à 1 à la clôture du premier chandelier, mais seulement à la clôture du deuxième chandelier. En effet, le premier ordre d'achat est passé à l'ouverture du deuxième chandelier.

```
BUY 1 SHARE AT MARKET
IF LONGONMARKET THEN
    long = 1
ENDIF
```

LongTriggered

La commande **LONGTRIGGERED[n]** permet de connaître si une ouverture de position longue a eu lieu sur le chandelier n

`LongTriggered[nième chandelier précédent]`

Note: Il est possible d'utiliser LongTriggered sans l'associer à un numéro de barre défini entre crochet. Dans ce cas, le programme considérera la barre du chandelier en cours de calcul, comme si vous aviez écrit : LongTriggered=LongTriggered[0]

Exemple :

```
//On rouvre une position longue uniquement si nous n'avons pas eu une ouverture/fermeture
de position sur le même chandelier :
IF NOT LONGONMARKET AND LongTriggered = 0 THEN
    Buy AT MARKET
ENDIF
```

ShortTriggered

La commande **SHORTTRIGGERED[n]** permet de connaître si une ouverture de position courte a eu lieu sur le chandelier n

`ShortTriggered[nième chandelier précédent]`

Note: Il est possible d'utiliser ShortTriggered sans l'associer à un numéro de barre défini entre crochet. Dans ce cas, le programme considérera la barre du chandelier en cours de calcul, comme si vous aviez écrit : ShortTriggered=ShortTriggered[0]

Exemple :

```
//On rouvre une vente à découvert uniquement si nous n'avons pas eu une ouverture/ferme-
ture de position sur le même chandelier :
IF NOT SHORTONMARKET AND ShortTriggered = 0 THEN
    SellShort AT MARKET
ENDIF
```

TradeIndex

La commande **TRADEINDEX(n)** permet d'accéder à la barre du n^{ième} ordre précédemment exécuté.

TRADEINDEX (n^{ième} ordre précédent)

Note: Il est possible d'utiliser TradeIndex sans l'associer à un numéro de barre défini entre parenthèses. Dans ce cas, le programme considérera la barre du dernier ordre passé, comme si vous aviez écrit : TradeIndex=TradeIndex(1). TradeIndex est généralement associé à Barindex.

Exemple :

```
// Si on est en position depuis au moins 3 barres :  
IF LONGONMARKET AND (BarIndex - TradeIndex) >= 3 THEN  
    SELL AT MARKET  
ENDIF
```

TradePrice

La commande **TRADEPRICE(n)** permet de faire appel au prix auquel une transaction a été exécutée. Sa syntaxe est la suivante :

TRADEPRICE (n^{ième} ordre précédent)

On peut spécifier entre parenthèses l'ordre auquel on se réfère. Si on ne spécifie pas l'ordre à la suite de TradePrice, le programme considérera le prix du dernier ordre exécuté, comme si vous aviez écrit : TradePrice=TradePrice(1)

Exemple :

```
// Si la clôture est supérieure au prix d'exécution de la dernière transaction majoré de 2%  
IF LONGONMARKET AND CLOSE > 1.02 * TRADEPRICE THEN  
    SELL AT MARKET  
ENDIF
```

PositionPerf

Cette commande donne accès à :

- La performance (ratio gains/coût de la position) de la n^{ième} dernière position clôturée, si n > 0 (frais de courtage non compris)
- La performance (ratio gains/coût de la position) de la position en cours, si n=0 (frais de courtage non compris)

Sa syntaxe est la suivante :

POSITIONPERF (n^{ième} position précédente)

Si n n'est pas renseigné, le programme considérera que n=0 : PositionPerf=PositionPerf(0)

Exemple :

```
// Achète si le trade précédent a fait au moins 20% de gains  
IF NOT ONMARKET AND PositionPerf(1) > 0.2 THEN  
    BUY 1000 CASH AT MARKET  
ENDIF
```

PositionPrice

La commande PositionPrice permet de connaître le prix moyen de la position en cours.

POSITIONPRICE

Le prix moyen d'une position est défini comme la somme des prix d'entrée pondérés par la quantité de chaque ordre. Seuls les renforcements de position influenceront sur le prix moyen de la position.

Cette instruction peut être utilisée avec des crochets (décalage) : POSITIONPRICE[1] retourne la valeur qu'avait PositionPrice à la clôture du chandelier précédent.

Exemple :

Vous achetez une action à 5€, puis vous renforcez votre position en rachetant une action à 10€, puis une autre à 15€.

Le PositionPrice est donc de $(5+10+15) / 3 = 10€$

Vous décidez alors d'alléger votre position en vendant une action pour 20€. La valeur de PositionPrice ne changera pas et sera toujours égale à 10€.

StrategyProfit

Cette instruction retourne les gains ou pertes (absolues, dans la devise de l'instrument et hors frais de courtage) effectivement réalisés depuis le début du système de trading. Les gains/pertes latents ne sont pas pris en compte.

Elle peut être associée à l'instruction QUIT pour limiter les pertes d'un système.

STRATEGYPROFIT

Il est aussi possible d'y ajouter des crochets: StrategyProfit[1] donnera par exemple le montant du profit à la clôture de la barre précédente.

Exemple :

```
IF STRATEGYPROFIT < -500 THEN
    QUIT
ENDIF
```

Note :

Pour rappel, le code de vos systèmes est **évalué à la clôture de chaque chandelier**. En reprenant l'exemple ci dessus, les pertes peuvent être supérieures à 500€ : si un gap intervient ou si une forte baisse se produit au sein d'un seul chandelier notamment. Il est donc conseillé d'utiliser STOP LOSS pour protéger les pertes sur la position, et ce bloc de code pour arrêter le système.

Avertissement : Les exemples présentés dans ce manuel sont à but pédagogique. Pour votre propre trading, vous êtes entièrement libre dans le choix de vos critères. Toutes les informations présentées sont à caractère "général" et ne sauraient, en aucun cas, représenter de quelconques informations ou conseils personnalisés ou une quelconque incitation à acheter ou vendre des instruments financiers. Les performances passées ne présagent pas des futurs résultats et ne sont pas constantes dans le temps. Tout système de trading peut vous exposer à un risque de perte supérieur à votre investissement initial.

Définition des paramètres d'exécution des systèmes de trading

Des paramètres additionnels peuvent être définis à l'aide de l'instruction [DEFPARAM](#).



Rappel au sujet de [DEFPARAM](#) : ils doivent toujours se situer au début de votre code

Cumul des ordres de même direction

Le paramètre "CumulateOrders" permet l'autorisation ou l'interdiction du cumul des ordres, dans le cadre d'une entrée en position sur le marché ou d'un renforcement de position. Par défaut, ce paramètre est actif ("True") lorsque vous créez un nouveau code directement par programmation : à chaque nouvelle barre, le système pourra donc ajouter une position à une position existante si les conditions d'entrée en position sont vérifiées. Si vous souhaitez entrer sur le marché grâce à des ordres stops ou limites, l'activation de "CumulateOrders" permet d'avoir plusieurs stops ou limites actifs en simultanément.

Pour s'assurer qu'un système n'augmentera pas la taille d'une position déjà ouverte, il est nécessaire d'insérer l'instruction suivante en tout début de code :

```
DEFPARAM CumulateOrders = False
```

La commande [DEFPARAM](#) reste active pendant toute la durée d'exécution de votre système de trading. Il n'est pas possible de changer le paramètre de cumul des ordres pendant l'exécution du système.

Exemples:

```
// Ce code achètera 1 action à chaque barre, avec un maximum de 3 positions en cours
DEFPARAM CumulateOrders = True
If CountOfPosition < 3 THEN
    Buy 1 shares at market
Endif
// Ce code achètera 1 action au prix de 2 et une action supplémentaire au prix de 3,
DEFPARAM CumulateOrders = True
If CountOfPosition < 2 THEN
    Buy 1 shares at 2 Limit
    Buy 1 shares at 3 Limit
Endif
// Ce code achètera 5 actions une seule fois
DEFPARAM CumulateOrders = False
Buy 5 shares at market
```

Plusieurs ordres de même direction peuvent être placés en simultanément pour sortir du marché, même lorsque "CumulateOrders" est paramétré sur "False".

Exemple:

```
// Ce code achètera 5 actions. On pourra vendre jusqu'à 3 actions si le prix croise à la
// baisse la MM40, Toute la position est liquidée si on réalise une perte d'au moins 10%.
DEFPARAM CumulateOrders = False
Buy 5 shares at market
If close CROSSES UNDER Average[40] THEN
    SELL 3 SHARES AT MARKET
    Set stop %Trailing 10
Endif
```

Note au sujet des stops et limites lorsque CumulateOrders est actif ("True") : Il est toujours possible de placer des stops de protection et des limites profits alors que ce paramètre est actif. Dans ce cas, le niveau des ordres stops et limites prendra pour référence le niveau moyen d'entrée de vos positions. Leurs niveaux seront recalculés à chaque modification de la taille de la position globale.

Exemple:

Si vous achetez une action à 10,00€, que vous placez un stop de protection de 10%, et une limite profit à 150%, les niveaux initiaux de ces ordres seront : Stop à 9,00€, limite à 25,00€. Si vous achetez une seconde action au prix de 20,00€, votre prix d'entrée moyen sera de 15€. En conséquence, les nouveaux niveaux de stops et limites seront : Stop à 13,50\$, limite à 37,50€ (position prise dans son ensemble).

Note sur les codes générés en mode "création assistée" : `CumulateOrders` est désactivé par défaut dans les codes issus du mode création assisté (l'instruction "`DEFPARAM CumulateOrders = false`" est insérée en début de code).

Précharger de l'historique

L'instruction `DEFPARAM PreLoadBars` définit le nombre maximum de barres préchargées avant le démarrage d'un système de trading. Cette instruction se révèle très utile lorsque votre système utilise des indicateurs nécessitant le chargement d'une certaine quantité d'historique afin de se calculer. Ce paramètre est défini sur 1000 par défaut. Il ne peut être ni inférieur à 0 ni supérieur à 10 000. Il est nécessaire au système de lancer un calcul sur au moins un chandelier d'historique afin d'initialiser votre code sur nos serveurs. Dès lors l'utilisation de `DEFPARAM preloadbars=0` aura le même effet que `DEFPARAM preloadbars=1`, il y aura un chandelier de préchargés afin de pouvoir lancer votre système de trading.

La valeur entrée pour ce paramètre correspond à un maximum, car le préchargement dépend de la quantité d'historique disponible pour l'instrument et l'horizon temporel sélectionnés.

Exemple:

```
DEFPARAM PreLoadBars = 300
a = (Close + Open) / 2
IF Close CROSSES OVER Average[250] (a) THEN
    BUY 1 SHARE AT MARKET
ENDIF
```

Prenons l'exemple d'une moyenne mobile de période 250. Si le paramètre `PreLoadBars` est défini à 300, la moyenne mobile possédera suffisamment de données pour se calculer au démarrage du système. Cela ne sera pas le cas si le paramètre est défini à 200 (250>200).

Comme mentionné plus haut, 300 est ici un nombre maximum. Si moins de 300 barres sont disponibles avant le démarrage du système, le préchargement s'effectuera sur le nombre de barres à disposition.

Dans la continuité de cet exemple, le "BarIndex" de la première barre suivant le démarrage du système sera égal à 300. Par contre, si aucune barre n'est préchargée, le "BarIndex" de la première barre suivant le démarrage du système sera égal à 0.

Attention : une affectation de variable précédée de "ONCE" sera exécutée qu'une seule fois à la première lecture de cette ligne de code (y compris sur l'historique chargé avec « preloadbars »)

Dans l'exemple ci-dessous, la stratégie ne passera donc aucun ordre (car tmp est à 1 pendant le preloadbars, et aucun ordre ne peut passer pendant cette période).

Exemple :

```
DEFPARAM PreLoadBars = 200
ONCE tmp = 1
IF tmp = 1 THEN
    BUY 1 SHARE AT MARKET
    tmp = 0
ENDIF
```

FlatBefore et FlatAfter

```
DEFPARAM FlatBefore = HHMMSS
DEFPARAM FlatAfter = HHMMSS
```

HHMMSS est l'heure détaillée où HH correspond aux heures, MM aux minutes et SS aux secondes.

Ces instructions vous donnent la possibilité d'annuler n'importe quel ordre en attente, de fermer n'importe quelle position ouverte et empêchent l'ajout éventuel d'ordres additionnels avant l'heure paramétrée (dans le cas de FlatBefore) ou après l'heure paramétrée (dans le cas de FlatAfter) dans le fuseau horaire défini pour la stratégie.

La valeur associée à l'instruction FlatBefore doit toujours être supérieure à l'heure d'ouverture du marché (ou de début de l'horaire personnalisé s'il est défini). La valeur associée à l'instruction FlatAfter doit être inférieure à l'heure de fermeture du marché (ou de fin d'horaire personnalisé s'il est défini). Dans le cas contraire, ces paramètres n'auront simplement pas d'effet. Si l'heure paramétrée n'est pas un multiple de l'unité de temps principale du système de trading (c'est à dire qu'elle survient à l'intérieur d'un chandelier en cours), alors l'instruction DEFPARAM FlatAfter prendra effet à la clôture de ce chandelier. L'instruction DEFPARAM FlatBefore sera quant à elle appliquée dès la clôture du chandelier précédent la barre contenant l'heure paramétrée.

Exemple:

```
DEFPARAM FlatBefore = 093000 // Annule n'importe quel ordre en attente, ferme n'importe
quelle position ouverte et empêche l'ajout éventuel d'ordres additionnels par le système
avant 09h30min00sec (dans le fuseau horaire de la stratégie)
DEFPARAM FlatAfter = 160000 // Annule n'importe quel ordre en attente, ferme n'importe
quelle position ouverte et empêche l'ajout éventuel d'ordres additionnels par le système
après 16h00min00sec (dans le fuseau horaire de la stratégie)
```

Réinvestir les gains (backtests seulement)

```
DEFPARAM NoCashUpdate = True
```

Si cette option est activée, le cash disponible n'est pas mis à jour avec les gains, les pertes et les frais de courtage

Exemple :

Capital initial 10 000 €, avec "NoCashupdate = True". L'investissement maximal sera limité à 10 000€ quels que soient les gains et les pertes réalisés tout au long de l'exécution du Backtest.

Remarque :

Les paramètres définis à l'aide de l'instruction DEFPARAM doivent être définis en début de code (après éventuels commentaires)

MinOrder et MaxOrder (backtests seulement)

```
DEFPARAM MinOrder = n
```

```
DEFPARAM MaxOrder = p
```

Cette option permet de bloquer les ordres dont la quantité (en lots, contrats ou actions) est inférieure à *n* ou supérieure à *p*.

Exemple :

```
DEFPARAM MinOrder = 100
```

```
BUY 1000 CASH AT MARKET
```

Si le dernier prix de l'instrument est au dessus de 10€, la quantité de l'ordre sera inférieure à 100 et l'ordre sera rejeté.

Appeler des d'indicateurs

Indicateurs ProRealTime

Toutes les fonctions que vous utilisez pour la création de vos indicateurs personnels sont également accessibles pour la création de vos systèmes de trading, y compris les indicateurs ProRealTime (cf. partie glossaire à la fin de ce manuel pour la liste complète des fonctions).

Nous vous conseillons également de vous référer au manuel ProBuilder pour plus d'informations sur ces fonctions.

La quantité d'historique nécessaire pour le calcul d'un indicateur dépend de la nature de cet indicateur.

Par exemple, on considère généralement que le calcul d'une moyenne mobile exponentielle de *N* périodes (ExponentialAverage[*N*]) nécessite 20**N* barres pour donner un résultat précis.

Si la date de début d'un backtest est très proche de la première date du graphique, le serveur fournit de l'historique supplémentaire afin que les systèmes de trading affichent des résultats dès la première barre.

Indicateurs personnalisés

Comme lors de la programmation d'indicateurs, votre système peut faire appel à des indicateurs personnalisés grâce à l'instruction CALL.

Exemple :

```
a,b = CALL "HistoMACD"[5,6] // 5 et 6 étant les paramètres passés en entrée de la fonction HistoMACD
```

Pour en savoir plus sur l'utilisation optimale des CALL, reportez-vous à la section dédiée aux conseils de programmation des systèmes de trading.

Astuces de programmation

Avertissement : Les exemples présentés dans ce manuel sont à but pédagogique. Pour votre propre trading, vous êtes entièrement libre dans le choix de vos critères. Toutes les informations présentées sont à caractère "général" et ne sauraient, en aucun cas, représenter de quelconques informations ou conseils personnalisés ou une quelconque incitation à acheter ou vendre des instruments financiers. Les performances passées ne présagent pas des futurs résultats et ne sont pas constantes dans le temps. Tout système de trading peut vous exposer à un risque de perte supérieur à votre investissement initial.

Le temps de calcul d'un système dépend de la complexité des indicateurs utilisés et de la façon dont ils sont appelés. Le paragraphe suivant propose donc quelques conseils simples pour optimiser votre code.

Réduire le nombre d'appels aux indicateurs ProRealTime

Si vous utilisez plusieurs fois le même indicateur ProRealTime dans des calculs différents, ne rappelez pas cet indicateur. Préférez toujours stocker l'indicateur dans une variable intermédiaire:

CODE NON-OPTIMAL	CODE OPTIMAL
<pre>IF NOT LONGONMARKET AND close > Average[40] (close) THEN BUY 1 SHARE AT MARKET ENDIF</pre>	<pre>avg40 = Average[40](close) IF NOT LONGONMARKET AND close > avg40 THEN BUY 1 SHARE AT MARKET ENDIF</pre>
<pre>IF NOT SHORTONMARKET AND close < Average[40] (close) THEN SELLSHORT 1 SHARE AT MARKET ENDIF</pre>	<pre>IF NOT SHORTONMARKET AND close < avg40 THEN SELLSHORT 1 SHARE AT MARKET ENDIF</pre>

Ce conseil est aussi valable lorsque vous voulez utiliser le même indicateur plusieurs fois mais avec un décalage différent.

CODE NON-OPTIMAL	CODE OPTIMAL
<pre>a = ExponentialAverage[40](close) b = ExponentialAverage[40](close[1]) c = ExponentialAverage[40](close) IF a > b THEN BUY 1 SHARE AT MARKET ENDIF IF a < c[1] THEN SELLSHORT 1 SHARE AT MARKET ENDIF</pre>	<pre>a = ExponentialAverage[40](close) IF a > a[1] THEN BUY 1 SHARE AT MARKET ENDIF IF a < a[1] THEN SELLSHORT 1 SHARE AT MARKET ENDIF</pre>

Appels d'indicateurs personnalisés

L'appel d'indicateurs personnalisés via l'instruction **CALL** est plus coûteux en temps de calcul que l'appel d'indicateurs ProRealTime. Pour les indicateurs ProRealTime, nous savons par avance quels calculs sont nécessaires et nous avons la maîtrise de leur exécution. Ce n'est pas le cas des indicateurs personnalisés, dont la programmation est propre à chaque utilisateur.

Pour améliorer la vitesse d'exécution d'un système, il est donc important d'optimiser l'utilisation de l'instruction **CALL** dans son code.

Limiter le nombre d'appels identiques :

Comme pour les indicateurs ProRealTime, il est conseillé de limiter le nombre d'appel aux indicateurs personnalisés au sein d'un même code :

CODE NON-OPTIMAL	CODE OPTIMAL
<pre>myindic1 = CALL "Ma Fonction" IF NOT LONGONMARKET AND close > myindic1 THEN BUY 1 SHARE AT MARKET ENDIF</pre>	<pre>myindic = CALL "Ma Fonction" IF NOT LONGONMARKET AND close > myindic THEN BUY 1 SHARE AT MARKET ENDIF</pre>
<pre>myindic2 = CALL "Ma Fonction" IF NOT SHORTONMARKET AND close < myindic2 THEN SELLSHORT 1 SHARE AT MARKET ENDIF</pre>	<pre>IF NOT SHORTONMARKET AND close < myindic THEN SELLSHORT 1 SHARE AT MARKET ENDIF</pre>

Limiter le nombre d'appels imbriqués :

Si vous souhaitez utiliser un indicateur personnalisé dans votre code, vérifiez qu'il n'utilise pas lui-même l'instruction CALL.

Les appels d'indicateurs personnalisés imbriqués les uns dans les autres sont très coûteux en temps de calcul. Si vous le pouvez, préférez toujours ré-insérer le code des indicateurs personnalisés que vous souhaitez utiliser dans votre système pour qu'il n'appelle que des indicateurs ProRealTime.

CODE NON-OPTIMAL	CODE OPTIMAL
<p>Code du système de trading :</p> <pre>myindic = CALL "MyDCLOSEMix LinearReg" IF NOT longonmarket AND close > myindic THEN BUY 1 SHARE AT MARKET ENDIF IF NOT shortonmarket AND close < myindic THEN SELLSHORT 1 SHARE AT MARKET ENDIF</pre> <p>Code de "MyDCLOSEMix LinearReg" :</p> <pre>dayclosemix = CALL "MyDCLOSEMix" RETURN LinearRegression[5](dayclosemix)</pre> <p>Code de "MyDCLOSEMix" :</p> <pre>mix = (DClose(0) + 3.5 * DClose(1) + 4.5 * DClose(2) + 3 * DClose(3) + 0.5 * DClose(4) - 0.5 * DClose(5) - 1.5 * DClose(6)) / 10.5 RETURN mix</pre>	<p>Code du système de trading :</p> <pre>dayclosemix = (DClose(0) + 3.5 * DClose(1) + 4.5 * DClose(2) + 3 * DClose(3) + 0.5 * DClose(4) - 0.5 * DClose(5) - 1.5 * DClose(6)) / 10.5 myindic = LinearRegression[5](dayclosemix) IF NOT longonmarket AND close > myindic THEN BUY 1 SHARE AT MARKET ENDIF IF NOT shortonmarket AND close < myindic THEN SELLSHORT 1 SHARE AT MARKET ENDIF</pre>

Limiter le nombre de conditions imbriquées :

Le conseil suivant est valable pour toutes les instructions conditionnelles (IF...THEN...ENDIF). Dans un souci d'optimisation du temps de calcul, il est toujours préférable de travailler avec une instruction vérifiant N conditions plutôt qu'avec N instructions :

CODE NON-OPTIMAL	CODE OPTIMAL
<pre> IF CLOSE >= 0.0014 THEN IF CLOSE <= 0.0047 THEN IF INTRADAYBARINDEX >= 5 THEN IF INTRADAYBARINDEX <= 20 THEN IF NOT SHORTONMARKET THEN BUY 1 SHARES AT MARKET ENDIF ENDIF ENDIF ENDIF ENDIF ENDIF ENDIF ENDIF </pre>	<pre> IF CLOSE >= 0.0014 AND CLOSE <= 0.0047 AND INTRADAYBARINDEX >= 5 AND INTRADAYBARINDEX <= 20 AND NOT SHORTONMARKET THEN BUY 1 SHARES AT MARKET ENDIF </pre>

Utilisation des boucles FOR :

L'utilisation de boucles FOR est parfois indispensable, mais il est mieux d'éviter de les utiliser quand cela est possible, car elles augmentent le temps de calcul.

Voici quelques exemples typiques où l'utilisation de boucle n'est pas nécessaire :

```

//Si la condition c1 a été au moins une fois vraie au cours des n derniers chandeliers :
IF HIGHEST[n](c1) = 1 THEN
  ...
//Si la condition c1 a toujours été vraie au cours des n derniers chandeliers :
IF LOWEST[n](c1) = 1 THEN
  ...
//Nombre de fois où la condition c1 a été vérifiée au cours des n derniers chandeliers :
num = SUMMATION[n](c1)

//Nombre de barres depuis que c1 a été vraie :
IF c1 THEN
  lastoccurrence = barindex
ENDIF
timesince = barindex - lastoccurrence

//Le max des variables (a,b,c,d,e,f,g) :
top = MAX(a , MAX(b , MAX(c , MAX(d , MAX(e , MAX(f , g) ) ) ) ) )

```

ProBacktest - La simulation des systèmes de trading

L'onglet "ProBacktest" de la fenêtre de création de systèmes de trading permet de configurer les paramètres de courtage qui seront utilisés pour la simulation du système :

ProBacktest

ProOrder AutoTrading

EURUSD +

Spot EUR/USD - 1 heure

Capital initial : 10000 \$

Taille lot : 1000 (Forex seulement)

Estimation des frais de courtage

Frais par ordre : 0 \$ / ordre

Min: 0 \$ / ordre

Max: 0 \$ / ordre

Spread : 2 pips

Personnaliser les variables

Période de la simulation

Début : Première date affichée

Fin : Dernière date (temps réel)

Critère d'optimisation : Gain

ProBacktest en mode tick par tick

Maintenir la fenêtre ouverte

Backtester mon système de trading

- Instrument(s)
- Graphique sur lequel appliquer le backtest
- Capital initial
- Taille du lot (forex)
- Paramètres de courtage
- Modèle de variable d'optimisation
- Période de simulation
- Critère d'optimisation

Gestion du capital / Money Management

Capital initial

Cette section permet de renseigner le capital alloué au système de trading. L'investissement maximal autorisé pendant l'exécution du système en dépend.

Pendant l'exécution du backtest, les frais de courtage, ainsi que les gains et pertes viendront l'augmenter ou le diminuer (sauf si le paramètre **NoCashUpdate** est activé). Reportez-vous au paragraphe [Réinvestir les gains](#) pour en savoir plus. Le capital disponible pour les systèmes en trading automatique correspond au capital disponible dans le portefeuille.



Si votre système de trading ne prend aucune position, pensez à augmenter le montant du Capital Initial.

Frais de courtage et spread

Vous pouvez ajuster ces paramètres pour refléter précisément les frais de votre courtier. Tout type de frais de courtage peuvent être appliqués à tous les types d'instruments. Les types de frais possibles sont :

- Montant fixe par ordre : montant fixe dans la devise de l'instrument appliqué chaque fois qu'un ordre est exécuté. Vous pouvez aussi spécifier une valeur minimale ou maximale par ordre.
- Pourcentage : pourcentage de la transaction (dans la devise de l'instrument) appliqué chaque fois qu'un ordre est exécuté.
- Montant par lot : montant fixe (dans la devise de l'instrument) appliqué par lot ou contrat exécuté.
- Taille lot (Forex seulement) : taille minimale d'un ordre sur le Forex. Chaque quantité d'ordre utilisée dans les instructions BUY/SELL est multipliée par la taille de lot.
- Spread (en points ou pips) : différence de prix entre meilleur Achat (Bid) et meilleure Vente (Ask).
-

EURUSD +

Spot EUR/USD - 1 heure

Capital initial : 10000 \$

Taille lot : 1000 (Forex seulement)

Estimation des frais de courtage

Frais par ordre : 0 \$ / lot

Min: 0 \$ / ordre

Max: 0 % de la transaction

Spread : 2 pips

Période de la simulation

Début : Première date affichée

Fin : Dernière date (temps réel)

Futures :

Les frais de courtage sont définis en commission par lot et par transaction.

La marge, ou "Deposit par lot" est le cash nécessaire pour l'achat d'un contrat. La valeur d'un point est automatiquement calculée par la plateforme en fonction du future concerné par le système de trading.

Voici les valeurs des points des principaux futures :

NOM DU FUTURE	VALEUR 1 POINT
FCE CAC 40	10€
DAX	25€
DJ Eurostoxx 50	10€
BUND	10€
Euro FX	12,5\$
Mini S&P 500	50\$
Mini Nasdaq 100	20\$
Mini Dow	5\$

Cette information est disponible dans notre langage grâce au mot-clé POINTVALUE.

Forex :

Les paramètres de courtage permettent de définir la taille des lots, le spread et la marge (effet de levier) à appliquer à chaque ordre.

Exemple avec l'EURUSD :

Taille d'un lot : 100 000

Spread : 2 pips

Marge : 5 % (=Effet de levier x20)

L'instruction `BUY 1 LOT AT MARKET` sur l'EURUSD aura pour effet d'acheter un lot de 100 000, avec un spread de 2 pips (soit 0,0002). L'effet de levier étant de 20 (marge de 5%), il est nécessaire de posséder 5 000€.

Actions :

Les frais de courtage sont définis en frais par ordre, en € ou % de la transaction. Il est également possible de définir des frais de courtage minimum par transaction.

Sur les actions, les tailles de position et les engagements par transaction sont définis en pourcentage du capital, ou dans la devise de l'instrument. Le Risk Management permet à la fois de brider le montant investi lors du passage d'un ordre, et d'appliquer un effet de levier.

Exemple :

- Frais de courtage par ordre : 10€
- Marge : 20 %

Avec cette configuration, chaque ordre passé pourra avoir un effet de levier de 5.

Optimisation des variables

Dans cette section, vous pouvez choisir d'optimiser des variables. Pour un instrument et horizon temporel donné, cette fonction permet de tester différentes combinaisons de variables, pour déterminer celles donnant les meilleures performances.

Pour en savoir plus et voir un exemple concret, vous pouvez visionner la vidéo "[Money management, stops et optimisation](#)".

Le résultat de l'optimisation est présenté dans un "Rapport d'optimisation". Vous connaîtrez les statistiques des meilleures combinaisons de variables testées et pourrez ainsi déterminer celle que vous devrez utiliser afin de rendre votre système de trading optimal.

Voici un exemple de système avec optimisation des période des moyennes mobiles n et m :

```
AVGm = ExponentialAverage[m] (Close)
```

```
AVGn = ExponentialAverage[n] (Close)
```

```
IF AVGm CROSSES OVER AVGn THEN
```

```
    BUY 100 SHARES AT MARKET
```

```
ENDIF
```

```
IF AVGm CROSSES UNDER AVGn THEN
```

```
    SELL 100 SHARES AT MARKET
```

```
ENDIF
```

Les variables n et m peuvent être définies en cliquant sur le bouton "Ajouter" dans la section "Optimisation des variables".



La fenêtre suivante apparaît alors :

Définition des variables - Mon backtest

Nom dans le programme :

Modèle par défaut Nouveau modèle +

n Valeur fixe :

Optimisation : Min Valeur maximale Pas (intervalle)

Nb de combinaisons :

Walk Forward

Type : Ancré Non ancré

Occurrences :

Période dans l'échantillon (optimisation) %

Période hors-échantillon (test) : %

[En savoir plus](#)

- **"Nom dans le programme"** représente le nom que la variable prend dans notre code (ici n et m).
- **"Valeur fixe"** sert à fixer la variable sur une valeur définie
- **"Valeur min"/"Valeur max"** représentent les bornes entre lesquelles la variable oscille,
- **"Pas"** représente l'intervalle minimum entre chaque valeur testée lors de l'optimisation.

Voici un exemple de rapport d'optimisation :

Gain	% Gain	Transactions	% Gagnant	Gain moyen
2 581,00 \$	+25,81 %	4041.0	+52,64 %	0,64 \$
2 326,60 \$	+23,27 %	1799.0	+56,48 %	1,29 \$
1 486,60 \$	+14,87 %	6461.0	+47,05 %	0,23 \$
1 140,10 \$	+11,40 %	3459.0	+51,60 %	0,33 \$
457,30 \$	+4,57 %	3973.0	+57,51 %	0,12 \$
-63,60 \$	-0,64 %	1263.0	+57,64 %	-0,05 \$
-109,00 \$	-1,09 %	5250.0	+45,90 %	-0,02 \$
-109,00 \$	-1,09 %	5250.0	+45,90 %	-0,02 \$
-109,00 \$	-1,09 %	5250.0	+45,90 %	-0,02 \$
-109,00 \$	-1,09 %	5250.0	+45,90 %	-0,02 \$
-109,00 \$	-1,09 %	5250.0	+45,90 %	-0,02 \$
-109,00 \$	-1,09 %	5250.0	+45,90 %	-0,02 \$
-109,00 \$	-1,09 %	5250.0	+45,90 %	-0,02 \$
-109,00 \$	-1,09 %	5250.0	+45,90 %	-0,02 \$
-109,00 \$	-1,09 %	5250.0	+45,90 %	-0,02 \$
-109,00 \$	-1,09 %	5250.0	+45,90 %	-0,02 \$
-109,00 \$	-1,09 %	5250.0	+45,90 %	-0,02 \$
-109,00 \$	-1,09 %	5250.0	+45,90 %	-0,02 \$
-109,00 \$	-1,09 %	5250.0	+45,90 %	-0,02 \$
-109,00 \$	-1,09 %	5250.0	+45,90 %	-0,02 \$
-109,00 \$	-1,09 %	5250.0	+45,90 %	-0,02 \$

Optimisation terminée

Le rapport en exemple indique 5 statistiques pour chacune des combinaisons de variables testées. Ces statistiques sont les suivantes :

- **"Gain"**, désigne la plus-value réalisée par le système de trading. Cela correspond à la formule :

$$\text{Profit} = \text{Montant du capital final} - \text{Montant du capital initial}$$

Cette statistique permet d'évaluer de façon absolue le potentiel de gains avec le système défini (pour une combinaison de variables et une période d'étude données).

Note: les frais de courtage définis dans la section " Gestion de capital" sont pris en compte dans ce calcul.

- **"%Gain"**, représente le profit ou la perte en pourcentage. Il correspond à la formule :

$$\% \text{Gain} = 100 \times \text{Profit} / \text{Montant du capital initial}$$

Indique donc la performance relative d'un système configuré avec les valeurs d'une combinaison de variables donnée.

- **"Transactions"**, indique le nombre de positions qui ont été ouvertes au cours de ce système.
- **"% Gagnant"** désigne le pourcentage de positions gagnantes du système. Mathématiquement, il se définit par :

$$\% \text{ trades gagnants} = (100 \times \text{Nombre de trades gagnants}) / \text{Nombre de trades}$$

- **"Gains moyen"** représente le gain moyen par position. Cette information est utile pour déterminer l'efficacité moyenne des ordres passés, notamment lorsque l'on ne souhaite pas passer beaucoup d'ordres. Elle se définit par :

$$\text{Gains moyens par position} = \text{Gains} / \text{Nombre de positions}$$

Note: les valeurs optimales des variables d'un système de trading peuvent changer, pour une même valeur, selon les unités de temps ou l'historique utilisé.

Définition de la période d'exécution du backtest

Période de la simulation



Début : Première date affichée

Fin : Dernière date (temps réel)

Période prédéfinie	Période personnalisée																																																																																																		
<input checked="" type="radio"/> Première date affichée <input type="radio"/> lun. 20 oct. 1997 00:00	<input checked="" type="radio"/> Dernière date (temps réel) <input type="radio"/> jeu. 15 déc. 2022 16:19																																																																																																		
<table border="1"> <thead> <tr> <th>lun.</th> <th>mar.</th> <th>mer.</th> <th>jeu.</th> <th>ven.</th> <th>sam.</th> <th>dim.</th> </tr> </thead> <tbody> <tr><td>29</td><td>30</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td></tr> <tr><td>6</td><td>7</td><td>8</td><td>9</td><td>10</td><td>11</td><td>12</td></tr> <tr><td>13</td><td>14</td><td>15</td><td>16</td><td>17</td><td>18</td><td>19</td></tr> <tr><td>20</td><td>21</td><td>22</td><td>23</td><td>24</td><td>25</td><td>26</td></tr> <tr><td>27</td><td>28</td><td>29</td><td>30</td><td>31</td><td>1</td><td>2</td></tr> <tr><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td></tr> </tbody> </table>	lun.	mar.	mer.	jeu.	ven.	sam.	dim.	29	30	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	1	2	3	4	5	6	7	8	9	<table border="1"> <thead> <tr> <th>lun.</th> <th>mar.</th> <th>mer.</th> <th>jeu.</th> <th>ven.</th> <th>sam.</th> <th>dim.</th> </tr> </thead> <tbody> <tr><td>28</td><td>29</td><td>30</td><td>1</td><td>2</td><td>3</td><td>4</td></tr> <tr><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>10</td><td>11</td></tr> <tr><td>12</td><td>13</td><td>14</td><td>15</td><td>16</td><td>17</td><td>18</td></tr> <tr><td>19</td><td>20</td><td>21</td><td>22</td><td>23</td><td>24</td><td>25</td></tr> <tr><td>26</td><td>27</td><td>28</td><td>29</td><td>30</td><td>31</td><td>1</td></tr> <tr><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td></tr> </tbody> </table>	lun.	mar.	mer.	jeu.	ven.	sam.	dim.	28	29	30	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	1	2	3	4	5	6	7	8
lun.	mar.	mer.	jeu.	ven.	sam.	dim.																																																																																													
29	30	1	2	3	4	5																																																																																													
6	7	8	9	10	11	12																																																																																													
13	14	15	16	17	18	19																																																																																													
20	21	22	23	24	25	26																																																																																													
27	28	29	30	31	1	2																																																																																													
3	4	5	6	7	8	9																																																																																													
lun.	mar.	mer.	jeu.	ven.	sam.	dim.																																																																																													
28	29	30	1	2	3	4																																																																																													
5	6	7	8	9	10	11																																																																																													
12	13	14	15	16	17	18																																																																																													
19	20	21	22	23	24	25																																																																																													
26	27	28	29	30	31	1																																																																																													
2	3	4	5	6	7	8																																																																																													

Ces champs vous permettent de définir les dates de début et de fin de votre backtest. La quantité de données utilisée pour un backtest est généralement limitée à l'historique affiché sur le graphique. Vous pouvez augmenter cette quantité d'historique via le menu déroulant en haut à gauche de chaque graphique. Il est nécessaire de charger la quantité d'historique que vous souhaitez avant de lancer votre backtest.

Dans le cas d'un backtest fait en "temps réel", les ordres apparaissent sur le graphique à chaque déclenchement de signal (achat, vente etc.). Il est également possible d'associer ces ordres à des popups ou à des sons depuis le menu "Option/configuration des alertes et des sons"

Si une date de fin est définie, les positions encore ouvertes à cette date seront fermées.

Remarque :

Si votre backtest met du temps à s'exécuter, vous pouvez réduire sa période d'exécution : le temps de calcul est proportionnel à la longueur de l'historique.


Après l'exécution d'un système de trading, les informations suivantes sont affichées :

- Évolution de la courbe gains & pertes à chaque chandelier
- Évolution des positions
- Rapport détaillé

Pour en savoir plus sur l'affichage des résultats de backtests, reportez-vous à l'Annexe A en fin de document.

Mode tick par tick :

Vous avez la possibilité de lancer votre backtest en mode « tick par tick » pour résoudre les problématiques de séquence d'exécution lorsque deux ordres peuvent être exécutés dans un même chandelier.

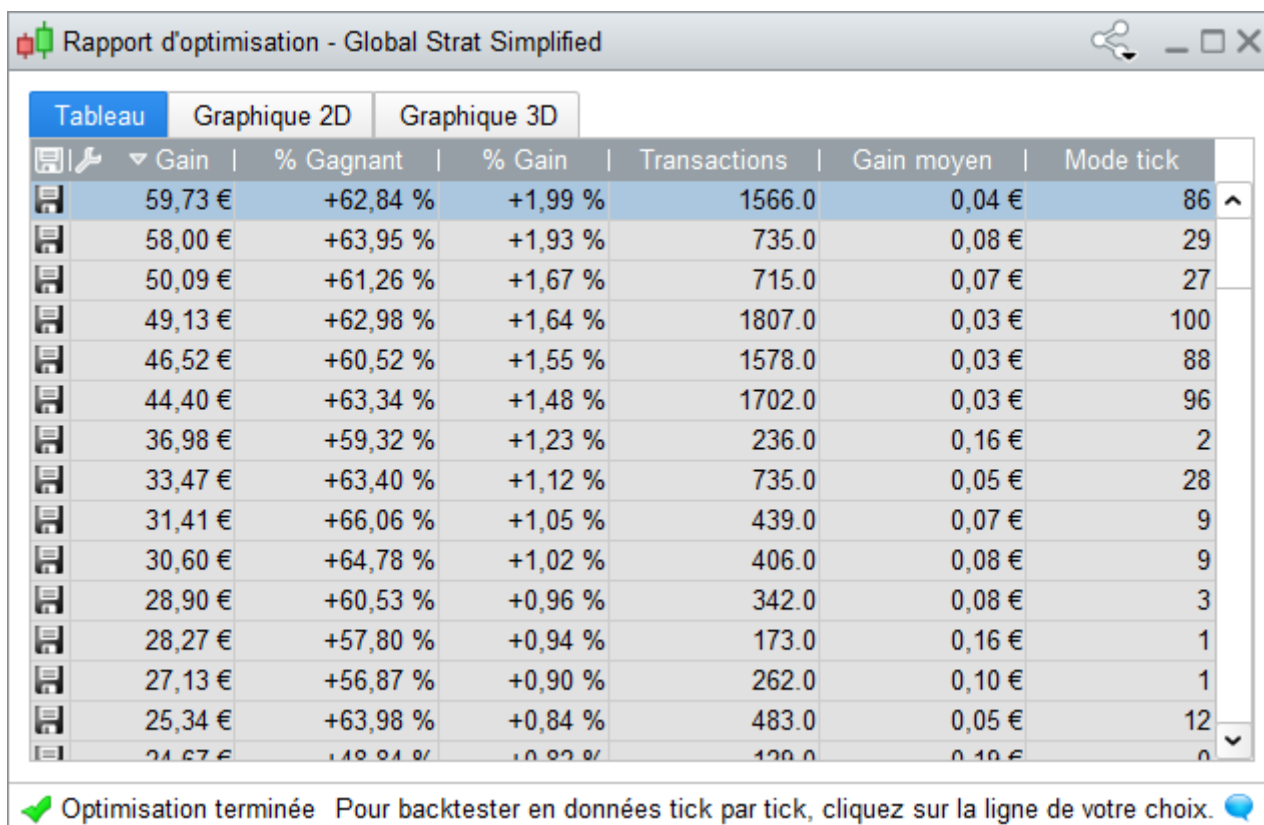
ProBacktest en mode tick par tick 

L'utilisation de cette option peut impacter le temps de calcul de votre système de trading. Plus le nombre de position nécessitant une vérification est important, plus l'impact sur les performances de calcul de votre code sera visible. Afin d'éviter des temps de calcul trop important, une limite de nombre de positions pouvant être vérifiées en mode "tick par tick" a été définie (500 par défaut, 2500 pour les plateformes premium).

Il faut aussi noter que l'utilisation de cette option peut impacter les résultats des prises de position de votre système de trading et ainsi modifier les performances globales de celui-ci.

Vous pouvez aussi utiliser cette option combinée avec les variables d'optimisations du module backtest, lorsque l'option est cochée, le backtest sera capable de compter le nombre de conflit entre une instruction **STOP** et **TARGET** dans un même chandelier.

Ainsi, vous verrez une nouvelle colonne dans votre rapport d'optimisation intitulée « Mode tick », elle représente le nombre de conflits pour chaque backtest.



	Gain	% Gagnant	% Gain	Transactions	Gain moyen	Mode tick
	59,73 €	+62,84 %	+1,99 %	1566.0	0,04 €	86
	58,00 €	+63,95 %	+1,93 %	735.0	0,08 €	29
	50,09 €	+61,26 %	+1,67 %	715.0	0,07 €	27
	49,13 €	+62,98 %	+1,64 %	1807.0	0,03 €	100
	46,52 €	+60,52 %	+1,55 %	1578.0	0,03 €	88
	44,40 €	+63,34 %	+1,48 %	1702.0	0,03 €	96
	36,98 €	+59,32 %	+1,23 %	236.0	0,16 €	2
	33,47 €	+63,40 %	+1,12 %	735.0	0,05 €	28
	31,41 €	+66,06 %	+1,05 %	439.0	0,07 €	9
	30,60 €	+64,78 %	+1,02 %	406.0	0,08 €	9
	28,90 €	+60,53 %	+0,96 %	342.0	0,08 €	3
	28,27 €	+57,80 %	+0,94 %	173.0	0,16 €	1
	27,13 €	+56,87 %	+0,90 %	262.0	0,10 €	1
	25,34 €	+63,98 %	+0,84 %	483.0	0,05 €	12
	24,67 €	+48,84 %	+0,82 %	120.0	0,10 €	0

Optimisation terminée Pour backtester en données tick par tick, cliquez sur la ligne de votre choix.

Après avoir cliqué sur une des lignes de votre rapport d'optimisation, le backtest associé sera lancé et résoudra chaque conflit pour savoir laquelle de vos instructions **STOP** et **TARGET** a été touchée en premier, vous donnant ainsi des résultats plus pertinent.

Note : les résultats affichés peuvent différer entre le rapport d'optimisation et les résultats du backtest car le mode tick par tick n'est pas appliqué directement lors l'optimisation mais seulement après avoir ouvert le backtest.

Personnalisation des plages horaires pour les backtests

La section « Fuseaux et plages horaires » de la fenêtre de « Réglages » permet de définir des plages horaires personnalisées pour les différents marchés.

Plages horaires personnalisées : si vous définissez des plages horaires réduites pour un marché, seules les données comprises dans cette plage horaire seront affichées sur les graphiques (et donc prises en compte dans le ProBacktest). Notez que les plages horaires réduites ne peuvent pas être définies à cheval sur deux jours de cotation distincts. Par exemple, sur un marché cotant 24h/24h, vous pouvez définir une plage horaire réduite entre 10h00 et 14h00, mais il n'est pas possible de définir une plage horaire entre 21h00 le premier jour et 09h00 du matin le jour suivant. Si un ordre est censé être placé à la clôture de la dernière barre de la plage horaire personnalisée, il sera alors placé à l'ouverture de la première barre de la plage personnalisée le lendemain.

Les constantes de prix $Dopen(x)/Dhigh(x)/Dlow(x)/Dclose(x)$ sur le chandelier courant ($x=0$) prennent en compte les plages d'horaires personnalisées que vous avez définies. Elles prennent les données officielles du marchés (sans plages horaires personnalisées) pour les chandeliers précédent ($x>0$).

L'utilisation de l'option d'application de ces réglages dans les unités de temps journalière et supérieure n'impacte pas ce comportement.

Notes concernant les données du week-end :

Certains des marchés cotant 24h/24h proposent l'option « Utiliser les cotations intraday pour construire les chandeliers journaliers ». Cette option n'est pas prise en compte par le ProBacktest, qui n'utilise que les chandeliers journaliers basés sur les horaires standard de marchés.

Certains marchés incluent des données du week-end (comme le Forex). Pour ces marchés, il est possible de cocher un boîtier permettant de cacher les données du week-end dans les graphiques, via le menu « Options/Fuseaux et plages horaires des graphiques intraday ». Les données du week-end sont toujours prises en compte pour les backtests. Concernant le Forex, les données du dimanche sont incluses dans le chandelier journalier du lundi (il n'existe pas de chandelier journalier pour le dimanche). Attention selon les conditions fixées par votre broker cette règle peut-être différente et le dimanche peut avoir son propre chandelier journalier.

Notes concernant les fuseaux horaires personnalisés :

Le code est toujours exécuté dans le fuseau horaire utilisateur. C'est à dire que toutes les instructions temporelles (Time, FlatAfter, FlatBefore) prendront en compte ce fuseau horaire dans leur calcul. Le fuseau horaire utilisateur est le fuseau horaire choisi pour le marché auquel l'instrument appartient, Il est configurable dans La section « Fuseaux et plages horaires » de la fenêtre de « Réglages ».

Par défaut, le fuseau horaire de l'instrument est celui de votre ordinateur.

Il est possible de changer le fuseau horaire d'un marché depuis La section « Fuseaux et plages horaires » de la fenêtre de « Réglages ». Au prochain lancement du backtest sur un instrument de ce marché, le nouveau fuseau horaire sera pris en compte.

Exemple : Sur Vodafone (sur le LSE dont le fuseau horaire officiel est GMT+1 en heure d'été ou BST), je configure mon graphique en heure de Paris (GMT+2 heure d'été, ou CET).

En 15 min, sur le chandelier de 11h45 heure de Londres (BST, GMT+1 en heure d'été), l'instruction Time va retourner 130000 (heure de cloture du chandelier 15 min de 11h45 BST : 12h00, transposé en heure CET : 13h00)

Les instructions concernées sont toutes les instructions temporelles intraday :

- Time et ses dérivés (Hour, Minute)
- Opentime et ses dérivés (OpenHour, OpenMinute)




- FlatAfter et FlatBefore
- Intradaybarindex (remis à zéro à l'ouverture du marché dans le fuseau horaire de l'utilisateur)

Les instructions temporelles en journalier ne sont pas concernées par ce changement;

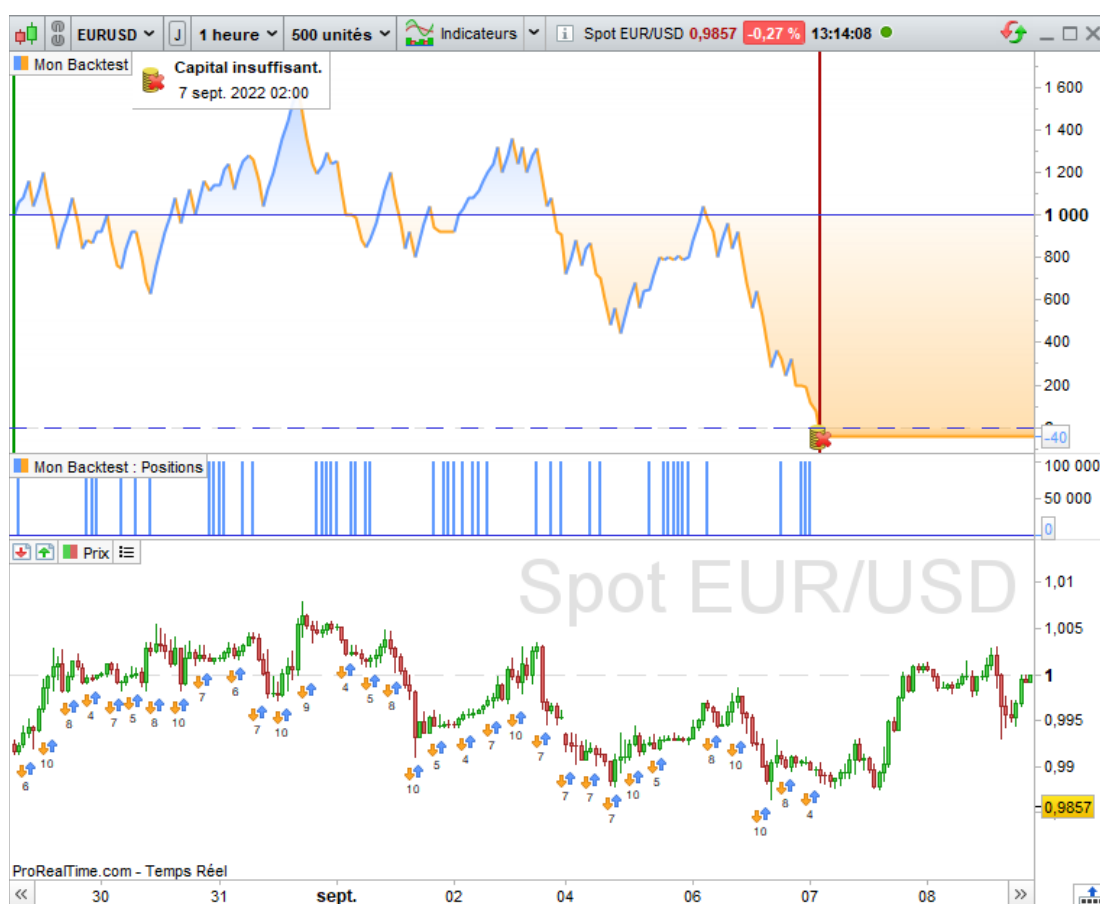
- DOpen,DHigh,DLow,DClose
- Date et ses dérivés (Year, Month,Day)
- DayOfWeek et Days

Raisons d'arrêt d'un ProBacktest

Un ProBacktest peut s'arrêter automatiquement pour l'une des raisons suivantes :

- Le backtest atteint l'heure de fin spécifiée dans la fenêtre de programmation. Dans ce cas, la fin du backtest est mise en évidence par une ligne verticale noire sur la courbe des gains et pertes.
- Il existe une instruction « QUIT » dans le code qui a été exécutée. Dans ce cas, la fin du backtest est mise en évidence par l'icône suivante : 
- Le capital disponible n'est plus suffisant pour couvrir les pertes (capital « estimé » est négatif). Dans ce cas, la fin du backtest est mise en évidence par l'icône suivante : 
- Un ordre est rejeté du fait de liquidités insuffisantes. Cet ordre apparaîtra dans la liste des ordres du rapport détaillé du backtest. La fin du backtest est mise en évidence par l'icône suivante : 

Voici un exemple de backtest arrêté suite à un capital insuffisant :



Affichage des valeurs des variables backtest

La fonction `GRAPH` permet de visualiser les valeurs de variables que vous utilisez dans votre programme backtest.

Cette instruction s'utilise de la façon suivante :

```
GRAPH myvariable AS "my variable name"
```

- `myvariable` est le nom de la variable dans le code
- `"my variable name"` est l'étiquette de la variable qui sera affichée sur le graphe

Il est également possible de définir une couleur de sortie grâce au paramètre optionnel `COLOURED` :

```
GRAPH myvariable COLOURED (r,g,b) AS "my variable name"
```

`r,g,b` sont des entiers entre 0 et 255 au format RGB.

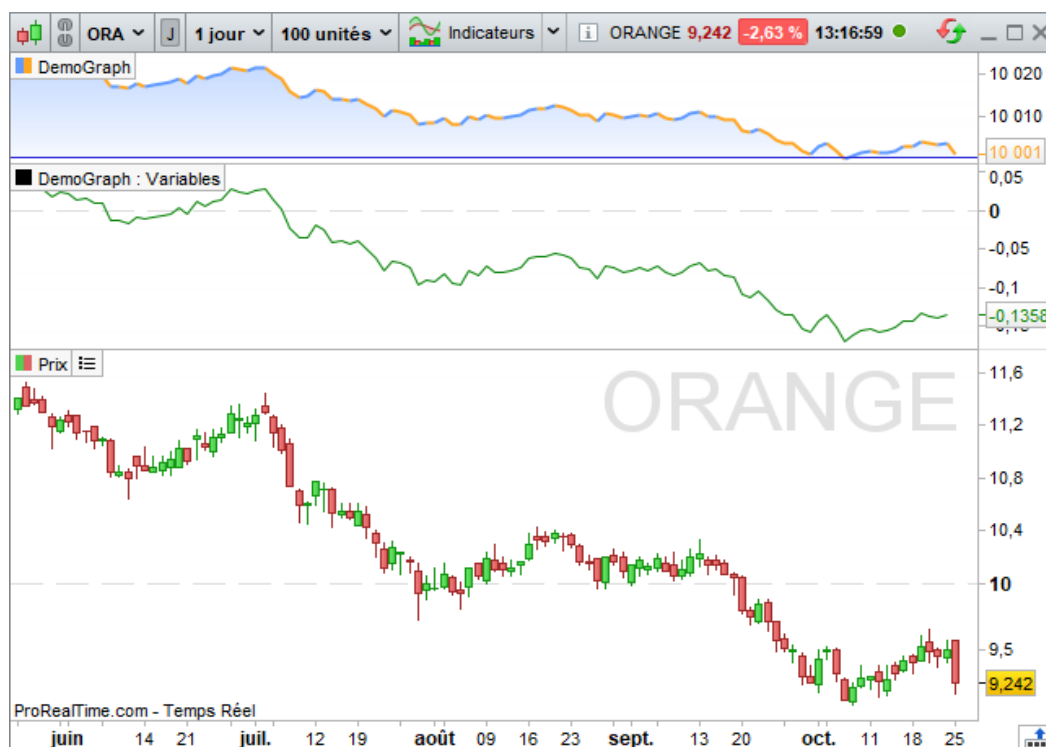
exemple : (255,0,0) pour une courbe rouge

ainsi que la transparence/opacité de la courbe :

```
GRAPH myvariable COLOURED (r,g,b,a) AS "my variable name"
```

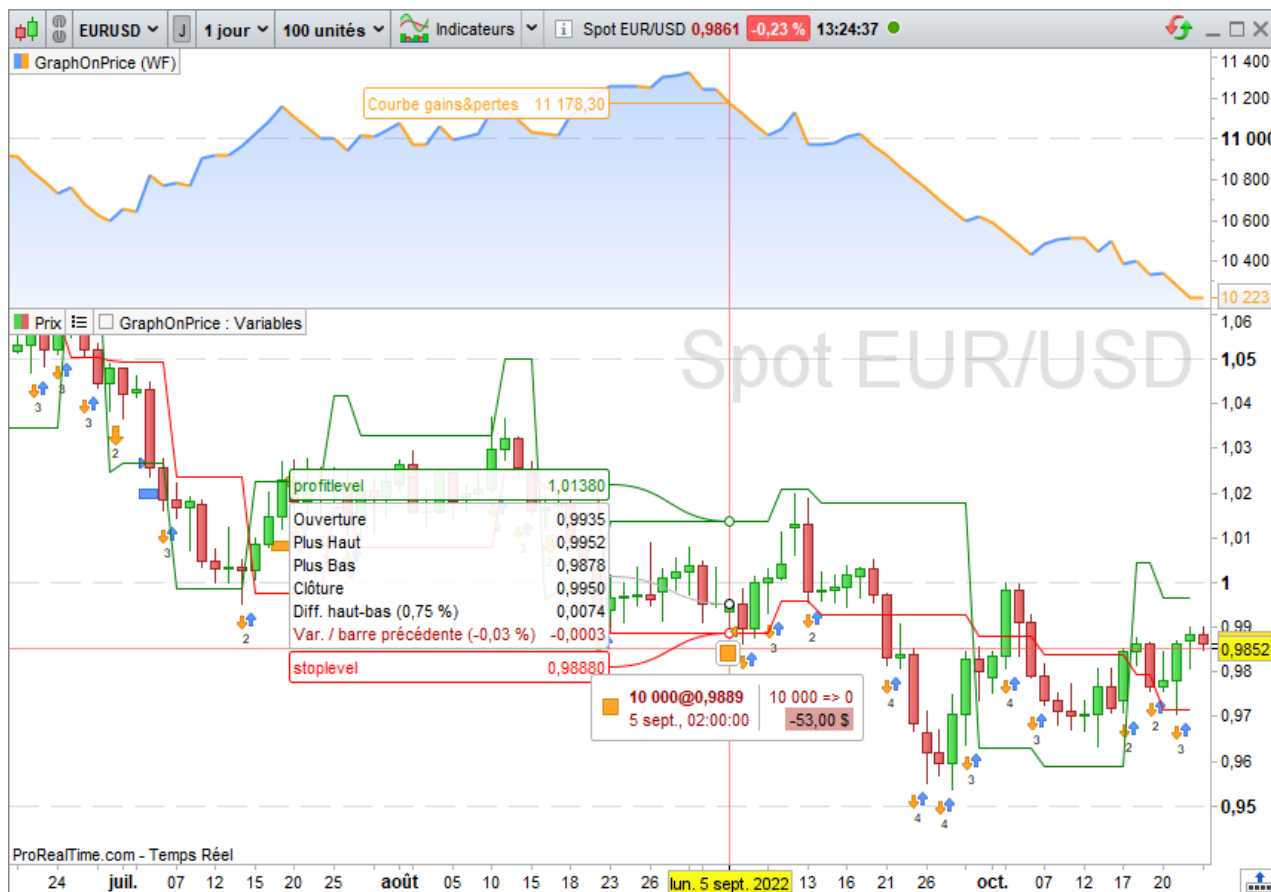
`a` est un entier compris entre 0 et 255 et indique la transparence (0 pour une courbe transparente, 255 pour une courbe opaque)

Un nouveau sous-graphe apparaît sous la courbe Gains/Pertes de votre backtest. Ce sous-graphe contient les valeurs de `myvariable` à chaque barre, comme dans l'exemple ci dessous :



Il est également possible de visualiser une ou plusieurs valeurs utilisées dans votre programme Backtest directement sur le graphique de prix avec l'instruction [GRAPHONPRICE](#)

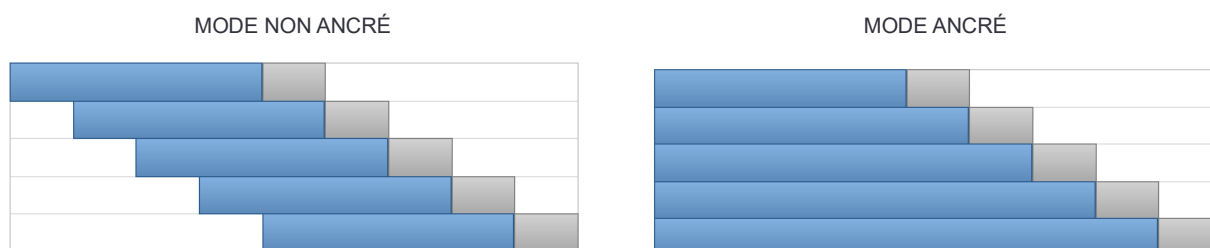
Dans l'exemple qui suit, les niveaux de Stop et de Profit sont affichés directement sur le graphique de prix permettant de suivre le placement des niveaux avec l'évolution du cours.



Note : Les instructions [GRAPH](#) et [GRAPHONPRICE](#) ne sont pas utilisables en Trading Automatique.

La méthode Walk Forward

L'analyse **Walk Forward** fait partie intégrante du processus de développement d'une stratégie de trading. Elle permet d'optimiser un système de trading en testant sa robustesse et sa stabilité au fil du temps. Tout d'abord, l'analyse optimise les variables définies sur une première période intitulée "période dans l'échantillon" ou "période d'optimisation". A l'issue de cette première étape, le meilleur jeu de variables est testé sur une période consécutive et non utilisée appelée "période hors échantillon" ou "période de tests". Ce processus automatisé peut être répété autant de fois que nécessaire tout en avançant dans le temps. Les périodes d'optimisations peuvent avoir des points de départ différents (mode non-ancré) ou un même point de départ (mode ancré).



En comparant ces 2 périodes, la méthode permet de :

- S'assurer que les performances générées par la stratégie sur les périodes de tests sont compatibles avec celles des périodes d'optimisation. Plus précisément, il s'agit d'éviter le risque de sur-optimisation d'un système en projetant la stratégie sur une période de temps non-utilisée.
- Vérifier le comportement d'une stratégie lorsque les conditions de marchés changent.
- Tester la robustesse de la stratégie sur les données du passé.

Afin de déterminer si une stratégie est robuste ou non, on utilise la mesure statistique du WFE (pour Walk Forward Efficiency). Indicateur qualitatif du processus d'optimisation, il compare le gain annualisé de la période de test par rapport au gain annualisé de la période d'optimisation. Cette mesure de robustesse est l'un des points centraux de l'analyse.

$$\text{Ratio WFE} = \frac{\text{Gain annualisé de la période de test}}{\text{Gain annualisé de la période d'optimisation}}$$

Le gain annualisé représente le gain réalisé sur une période déterminée exprimé en année. Il permet de comparer le gain de périodes différentes sur une même base. En effet, la période d'optimisation représente souvent 70% de la période totale de l'analyse, contre 30% pour la période de test. Pour mesurer la fiabilité de l'analyse il convient donc d'évaluer ces deux périodes sur une base commune.

Les différentes études menées sur cette méthode affirment que si au moins 3/5 des périodes de tests affichent un ratio WFE supérieur à 50%, une stratégie peut être considérée comme étant robuste.

A l'issue de la méthode Walk-Forward, il convient de porter l'analyse des résultats sur plusieurs points. Il est en effet nécessaire de vérifier la régularité des rendements issus des périodes post-optimisation. Si les résultats démontrent que la stratégie comporte un risque de sur-optimisation (les rendements issus des périodes de tests sont largement inférieurs aux rendements des périodes d'optimisation), la plateforme donne la possibilité d'ajuster différents paramètres (variables, niveaux de stops et limites, période de trading) et de relancer le processus Walk-Forward autant de fois que nécessaire pour obtenir une stratégie robuste.

Cas pratique

Cette section présente comment optimiser un système de trading en utilisant la méthode Walk Forward depuis une stratégie existante.

En premier lieu, cliquez sur la clé à molette en haut à gauche de la fenêtre ProBacktest & Trading Automatique.

Optimisation des variables : NUMBER[10;40][1], ORDERS[5;10][1], PROTECTION[3]

```

1 // Définition des paramètres du code
2 DEFPARAM CUMULATEORDERS = False // Cumul des positions désactivé
3
4 // Conditions pour ouvrir une position acheteuse
5 indicator1 = SMI[14,3,5](close)
6 indicator2 = Average[3](SMI[14,3,5](close))
7 c1 = (indicator1 CROSSES OVER indicator2)
8
9 indicator3 = close
10 indicator4 = Average[NUMBER](close)
11 c2 = (indicator3 >= indicator4)
12 indicator5 = close
13 IF c1 and c2 THEN
14     BUY ORDERS CONTRACT AT MARKET
15 ENDIF
16
17 // Conditions pour fermer une position acheteuse
18 indicator5 = SMI[14,3,5](close)
19 indicator6 = Average[NUMBER](SMI[14,3,5](close))
20 c3 = indicator5 CROSSES OVER indicator6
21
22 IF c3 then
23     SELL AT MARKET
24 ENDIF
25
26 //Stops et objectifs : entres vos stops et vos objectifs ici
27 SET STOP %LOSS PROTECTION
28

```

ProBacktest

ProOrder AutoTrading

AIRBUS - 1 jour

Capital initial : 10000 €

Estimation des frais de courtage

Frais par ordre : 0 € / ordre

Min: 0 € / ordre

Max: 0 € / ordre

Spread : 2 points

Personnaliser les variables

Période de la simulation

Début : Première date affichée

Fin : Dernière date (temps réel)

Critère d'optimisation : Gain

ProBacktest en mode tick par tick

Maintenir la fenêtre ouverte

Backtester mon système de trading

[Formations vidéo](#)

La fenêtre ci-dessous apparaît. Une fois vos variables définies, vous pouvez activer la méthode Walk Forward en cliquant sur le bouton "Actif". Différents paramètres peuvent être appliqués. Sélectionnez la méthode ancrée pour des points de départs différents ou la méthode non-ancré pour des points de départs simultanés.

Vous avez également la possibilité de définir le nombre de répétitions et l'historique des 2 périodes (une répétition consiste à optimiser les paramètres sur la période d'optimisation et de tester le meilleur couple sur la période de test).

Définition des variables - Stochastic Momentum

Nom dans le programme

Modèle par défaut Nouveau modèle +

NUMBER	<input type="radio"/> Valeur fixe : <input type="text" value="1"/>	<input checked="" type="radio"/> Optimisation : Min <input type="text" value="10"/> Valeur maximale <input type="text" value="40"/> Pas (intervalle) <input type="text" value="1"/>
ORDERS	<input type="radio"/> Valeur fixe : <input type="text" value="1"/>	<input checked="" type="radio"/> Optimisation : Min <input type="text" value="5"/> Valeur maximale <input type="text" value="10"/> Pas (intervalle) <input type="text" value="1"/>
PROTECTION	<input checked="" type="radio"/> Valeur fixe : <input type="text" value="3"/>	<input type="radio"/> Optimisation : Min <input type="text" value="1"/> Valeur maximale <input type="text" value="10"/> Pas (intervalle) <input type="text" value="1"/>

Nb de combinaisons : 186/100000

Walk Forward

Type : Ancré Non ancré

Occurrences :

Période dans l'échantillon (optimisation) %

Période hors-échantillon (test) : %

[En savoir plus](#)

La recherche suggère d'utiliser une période d'optimisation représentant 80% de la période de simulation et 20% pour la période de tests. Il est également recommandé d'effectuer le plus de tests possibles afin de fiabiliser l'analyse.

Une fois vos paramètres établis, fermez la fenêtre ci-dessus et lancez l'analyse Walk Forward en cliquant sur le bouton "Backtester mon système de trading".

```

1 // c1 = (indicators CROSSSES OVER indicators)
2
3
4
5
6
7
8
9 indicator3 = close
10 indicator4 = Average[NUMBER](close)
11 c2 = (indicator3 >= indicator4)
12 indicator5 = close
13 IF c1 and c2 THEN
14     BUY ORDERS CONTRACT AT MARKET
15 ENDIF
16
17 // Conditions pour fermer une position acheteuse
18 indicator5 = SMI[14,3,5](close)
19 indicator6 = Average[NUMBER](SMI[14,3,5](close))
20 c3 = indicator5 CROSSSES OVER indicator6
21
22 IF c3 then
23     SELL AT MARKET
24 ENDIF
25
26 //Stops et objectifs : entres vos stops et vos
27 //objectifs ici
28 SET STOP %LOSS PROTECTION

```

Frais par ordre : € / ordre

Min: € / ordre

Max: € / ordre

Spread : points

Personnaliser les variables

Période de la simulation

Début : Première date affichée

Fin : Dernière date (temps réel)

Critère d'optimisation :

ProBacktest en mode tick par tick

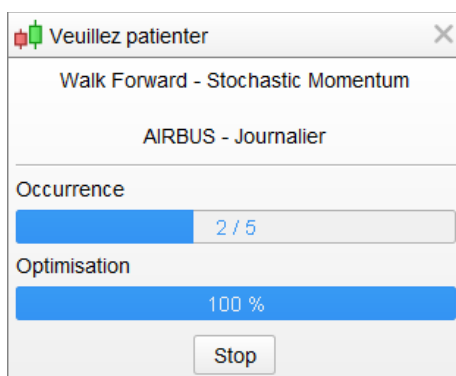
Maintenir la fenêtre ouverte

[Formations vidéo](#)

Durant l'analyse Walk Forward, la plateforme commence par optimiser la stratégie sur une première période dans l'échantillon et détermine le jeu de variables ayant généré la performance la plus élevée. Ensuite, la performance du même jeu de variable est testée sur un échantillon distinct.

Le processus est répété 5 fois. L'objectif est de déterminer si le jeu de variables jugé optimal génère une performance en ligne voire supérieure dans des conditions de marchés différentes.

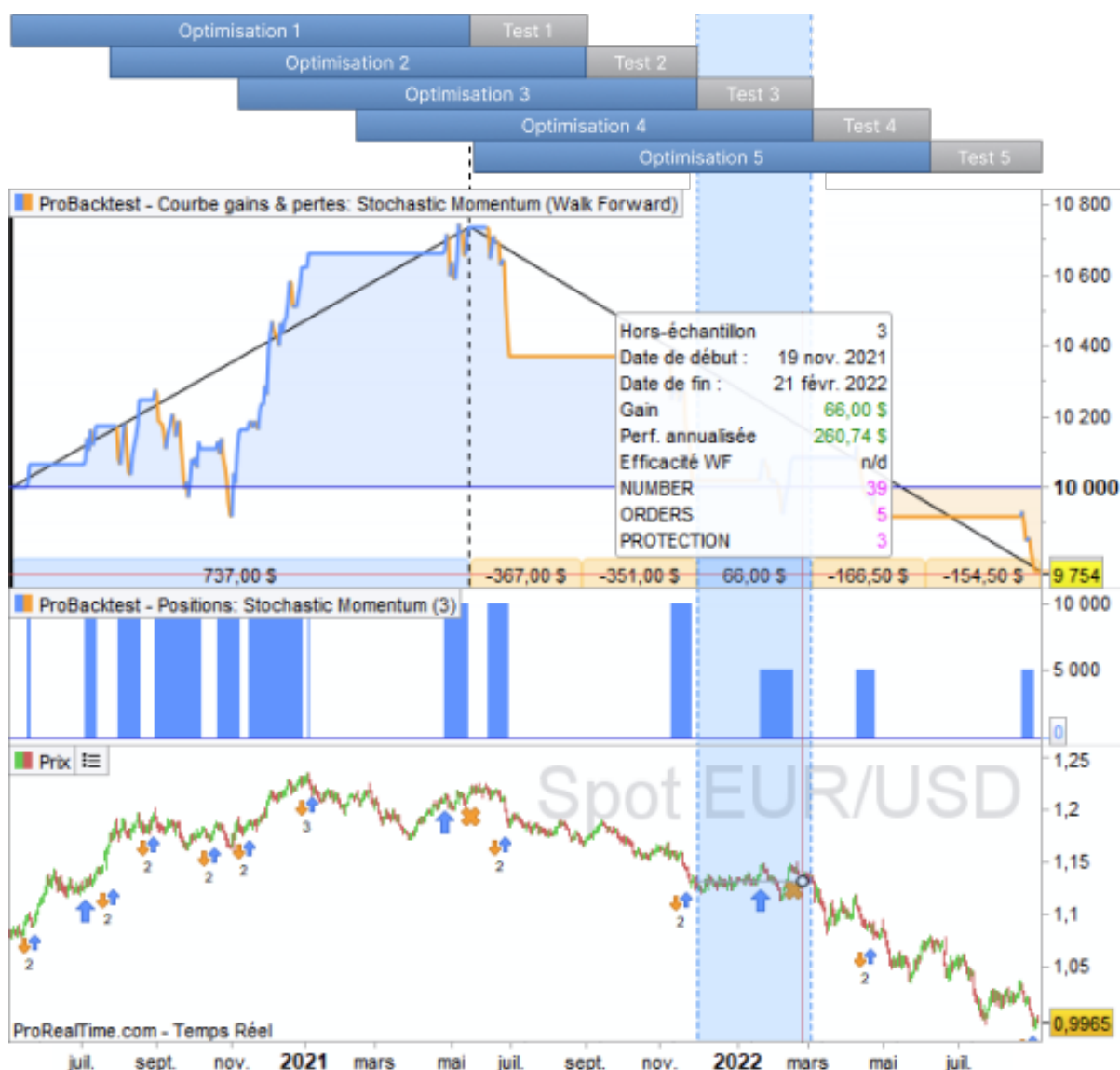
Selon le nombre de variables et le nombre d'occurrences définis, les calculs peuvent être plus ou moins complexes.



Après calculs, un graphique représentant la courbe des gains & pertes s'affiche ainsi qu'un rapport détaillé des performances du système. En plus des onglets courants, une nouvelle partie "Statistiques Walk Forward" présente la comparaison des différentes périodes de l'analyse pour chaque répétition.

Rapport détaillé									
ProBacktest (Walk Forward)									
Stochastic Momentum									
AIRBUS									
Statistiques Walk Forward									
Stochastic Momentum Journalier									
Début : 6 avr. 2020 [10 000,00 €]									
Actuel : 4 août 2022 [9 956,70 €]									
#	Éch. / Hors-É...	Début	Fin	Gain	Perf. annualis...	Efficacité WFI	Max drawdo...	Runup max	Variables
1	Échantillon	6 avr. 2020	22 avr. 2021	-90,10 €	-87,38 €		-146,45 €	0,00 €	x
	Hors-échantil...	23 avr. 2021	23 juil. 2021	35,50 €	140,46 €	n/d	-45,30 €	86,05 €	
2	Échantillon	13 juil. 2020	23 juil. 2021	-16,70 €	-16,26 €		-104,35 €	0,00 €	x
	Hors-échantil...	26 juil. 2021	26 oct. 2021	-17,20 €	-67,02 €	n/d	-17,20 €	4,70 €	
3	Échantillon	13 oct. 2020	26 oct. 2021	-13,40 €	-13,00 €		-82,80 €	0,00 €	x
	Hors-échantil...	27 oct. 2021	27 janv. 20...	46,50 €	183,47 €	n/d	-64,40 €	92,90 €	
4	Échantillon	18 janv. 20...	27 janv. 20...	85,80 €	83,21 €		-165,60 €	0,00 €	x
	Hors-échantil...	28 janv. 20...	3 mai 2022	-34,40 €	-136,11 €	-163,57 %	-61,70 €	27,30 €	
5	Échantillon	23 avr. 2021	3 mai 2022	209,00 €	203,46 €		-128,80 €	0,00 €	x
	Hors-échantil...	4 mai 2022	4 août 2022	16,40 €	63,91 €	+31,41 %	-65,00 €	98,20 €	

Les statistiques ci-dessus sont relatives au passé. Les performances passées ne présagent pas de l'avenir.



La plateforme agrège et présente les résultats de la 1^{ère} période d'optimisation (échantillon 1) et les 5 périodes de tests (hors échantillon).

En plaçant le curseur de la souris sur les bandes inférieures, la période correspondante est affichée en surbrillance avec les paramètres utilisés ainsi que la performance générée. Sur la capture ci-dessus, la souris est placée sur la 4^e période de test (période hors échantillon #4).

La meilleure combinaison des variables PROTECTION, ORDERS et NUMBER issue de la 1^{ère} phase d'optimisation (échantillon 4) a été appliquée sur la phase de test suivante (en surbrillance). Ces paramètres ont généré un gain de 2 250 € sur cette période.

La stratégie étudiée ici semble en premier lieu être robuste. Seules 2 périodes hors-échantillons sur 5 affichent des performances inférieures aux périodes dans l'échantillon. Pourtant, si le risque (représenté par l'indicateur Max drawdown) mesuré sur les phases de tests est en ligne avec celui mesuré lors des phases d'optimisations, les ratios WFE indiquent la présence d'un biais. Un ratio trop important peut en effet indiquer une sur-optimisation de la stratégie ou un mouvement anormal du marché. Si un événement économique impacte trop favorablement les résultats de la stratégie, il est important de noter que cet impact aurait pu être tout aussi défavorable si les marchés avaient réagi différemment face à cette actualité exceptionnelle. Il serait donc intéressant de réaliser un nombre plus important d'occurrences et de vérifier si ce biais persiste. Si tel est le cas, l'analyse Walk-Forward aura mis en évidence un risque de sur-optimisation de la stratégie. Il faut donc revoir ses paramètres initiaux.

ProOrder - Le Trading Automatique

Avertissement : Si vous mettez en œuvre un système de trading au moyen du service ProOrder, ce service enverra des signaux de manière automatique, selon les paramètres que vous avez déterminés, en vue de l'exécution des ordres correspondants sans qu'aucune validation individuelle de chaque ordre de votre part ne soit requise. Votre système sera exécuté automatiquement même lorsque votre ordinateur sera éteint. Il est de votre responsabilité de vous assurer que vous avez paramétré votre système de telle sorte qu'il ne conduise pas à la réalisation de pertes au-delà d'un montant que vous êtes prêt à accepter.

En tout état de cause, ProRealTime ne sera responsable des éventuelles pertes subies suite à l'exécution depuis de votre système de trading automatique.

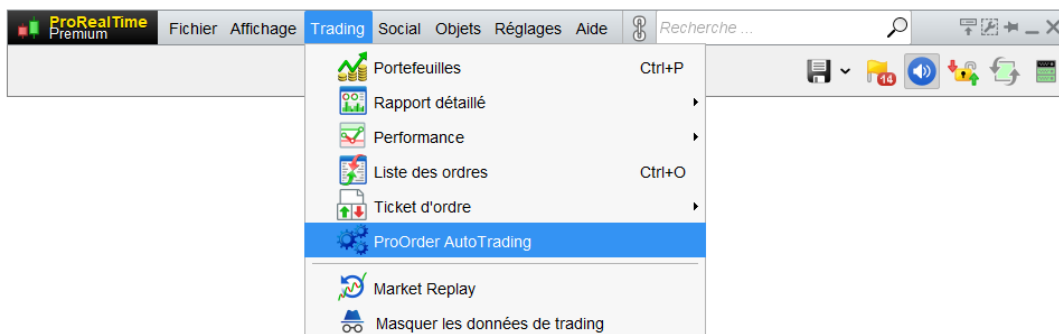
Cette partie du manuel expliquera comment passer du backtest de vos systèmes de trading à leur exécution en tant que systèmes de trading automatique.

- La première section détaille l'envoi d'un système de trading vers ProOrder et les étapes nécessaires à son exécution en trading automatique.
- La seconde section montre comment lancer un système de trading automatique et vérifier sa performance.
- La troisième section décrit les paramètres des systèmes de trading et leur conditions d'exécution.
- La quatrième section explique comment trading manuel et trading automatique interagissent dans la plateforme.
- La cinquième section fournit des informations spécifiques à l'exécution de plusieurs systèmes de trading automatique sur un même instrument.
- La dernière section contient une liste d'indicateurs qui ne peuvent pas être utilisés en trading automatique à cause de leur méthode de calcul.

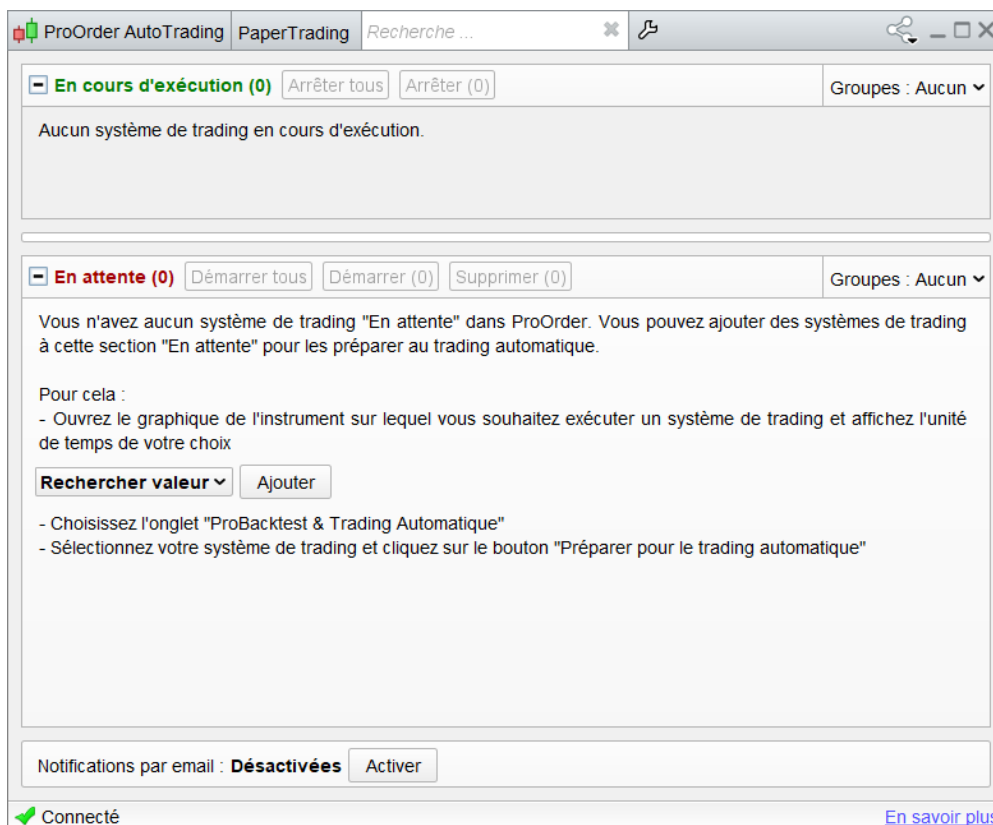
Nous vous conseillons de terminer la lecture du manuel avant de vous lancer dans l'exécution d'un système en trading automatique.

Préparer un système à l'exécution en trading automatique

Commencez par ouvrir la fenêtre ProOrder depuis le menu "Trading" :

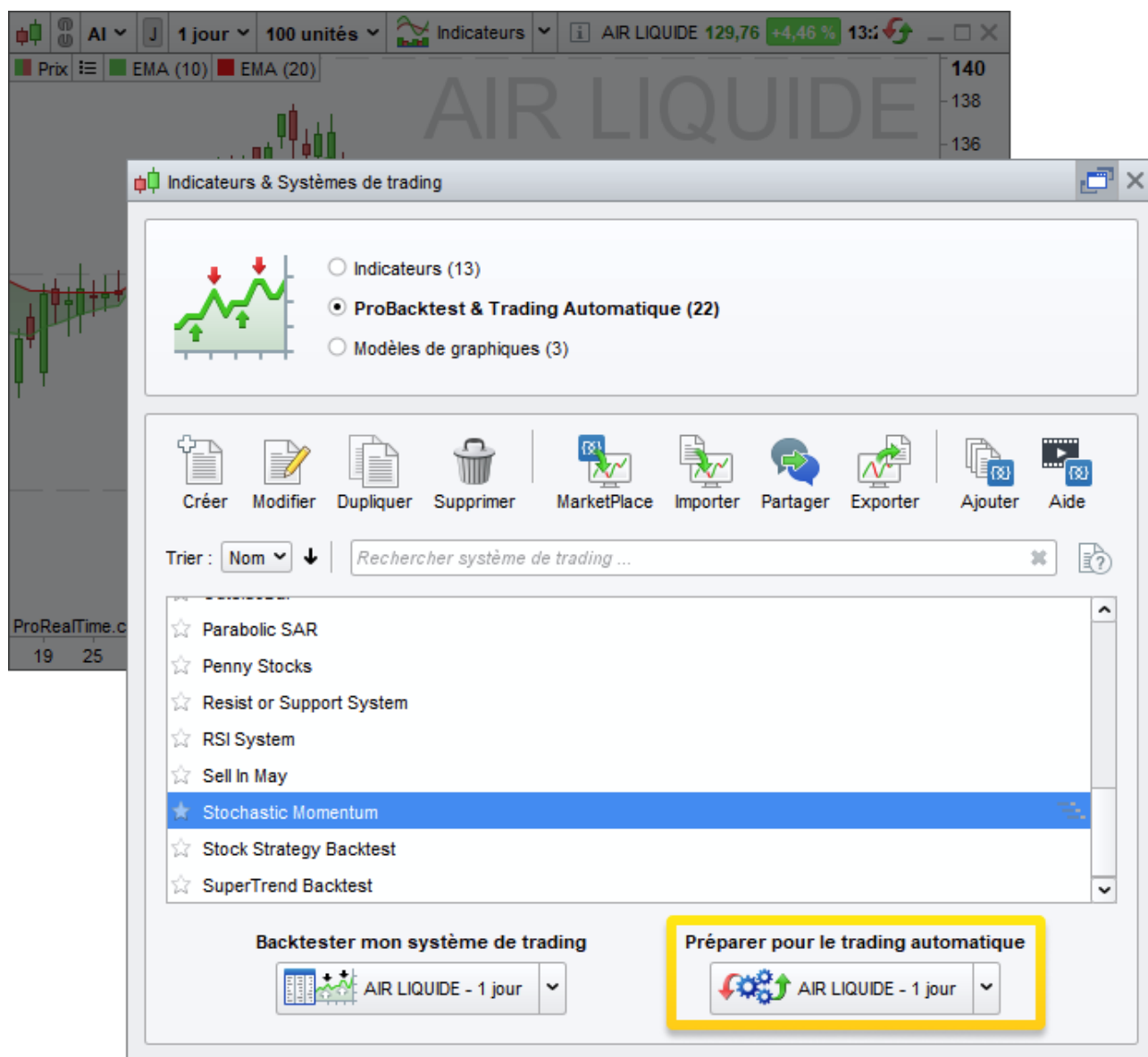


La fenêtre illustrée ci-dessous s'affiche. Elle contient les étapes à suivre afin de préparer un système pour l'exécution en trading automatique.



Choisissez une valeur puis cliquez sur le bouton « Ajouter ».

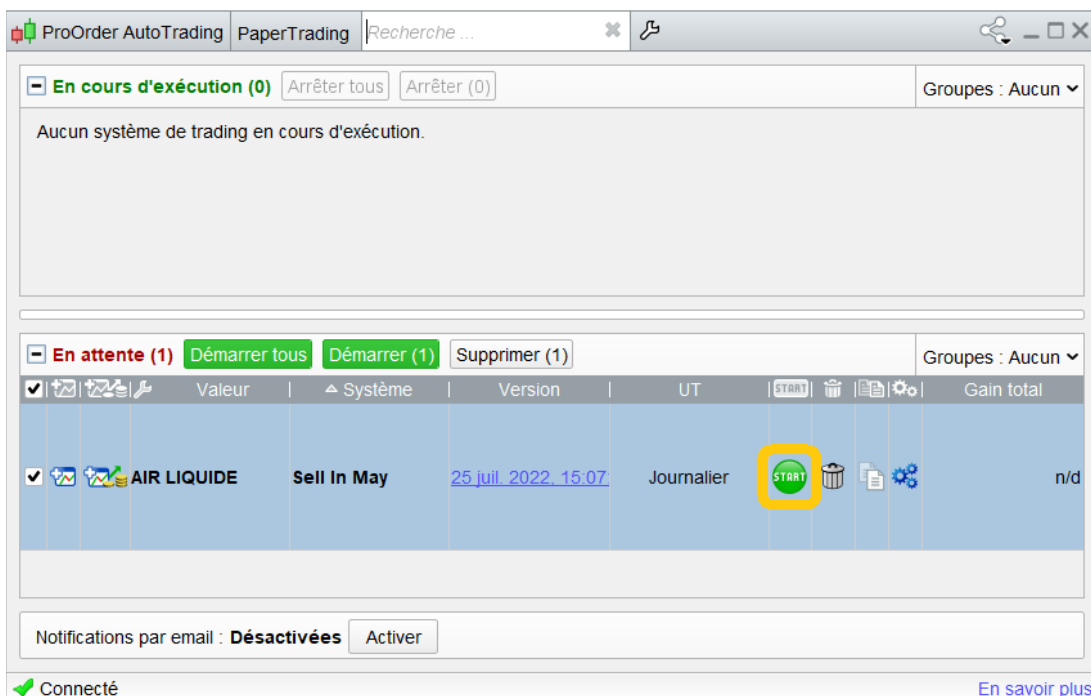
La fenêtre "Indicateurs et Systèmes de Trading" apparaît. Vous obtenez une liste de vos systèmes de trading automatique en cliquant sur la catégorie correspondante.



Sélectionnez le système de votre choix et cliquez sur "Préparer pour le trading automatique". Le système apparaîtra ainsi dans ProOrder sur la valeur choisie initialement.

Démarrer un système de trading avec ProOrder et voir sa performance

Une fois que vous avez ajouté un système de trading à ProOrder, vous pouvez définir son paramètre de taille maximum de position. Lancez ensuite votre système en cliquant sur le bouton "Start" :



Veillez à bien lire le contenu de la pop-up de confirmation : il vous sera demandé de confirmer l'exécution du système de trading. Il est important de comprendre que le paramètre de taille maximum de position a la priorité sur les quantités d'achat/vente définies dans le code du système. La taille de position maximum pour les futures et le Forex est définie en nombre de contrats ou nombre de lots. Cette taille de lot est définie par le broker et n'est donc pas modifiable.

Par exemple, si votre code comporte une instruction d'achat de 3 lots et que la limite de taille maximum de position est fixée à 1, l'ordre d'achat des 3 lots sera ignoré.

De même, si votre code prévoit successivement l'achat d'1 lot puis la vente à découvert de 3 lots, l'ordre de vente à découvert sera ignoré et vous resterez en position acheteuse d'1 lot. Il est recommandé de toujours vérifier le paramètre de taille maximum de position avant de lancer un système de trading.

Pour les actions, la taille maximum de position est exprimée en montant (frais de courtage non-inclus).

Une fois lancé, le système sera affiché dans la partie "En cours d'exécution" de la fenêtre, comme illustré ci-dessous.

The screenshot displays the ProOrder AutoTrading interface. At the top, there are tabs for 'ProOrder AutoTrading' and 'PaperTrading', along with a search bar. Below this, a section titled 'En cours d'exécution (1)' contains a table with one entry. The table has columns for 'Valeur', 'Système', 'Heure de démarrage', 'Version', 'UT', 'Position', 'Gain latent', and 'Gain total'. The entry is for 'Spot EUR/USD' using the 'EUR/USD Strategy 2.0' system, which started on 01/08/2022 at 14:36. The version is '28 juil. 2022_17:22:28', and it has been running for '2 minutes'. The position is '110 000', the latent gain is '32.00 \$', and the total gain is '962.10 \$'. There are buttons for 'Arrêter tous' and 'Arrêter (1)' at the top right of this section. Below the table, there is a section for 'En attente (0)' with buttons for 'Démarrer tous', 'Démarrer (0)', and 'Supprimer (0)'. A text block explains that no trading systems are currently in the 'En attente' state and provides instructions on how to add a system. At the bottom, there are controls for email notifications (currently 'Désactivées') and a validity period ending on '3 août 2022 16:00'. The interface also shows a 'Connecté' status and a link to 'En savoir plus'.

Valeur	Système	Heure de démarrage	Version	UT	Position	Gain latent	Gain total
Spot EUR/USD	EUR/USD Strategy 2.0	01/08/2022 14:36	28 juil. 2022_17:22:28	2 minutes	110 000	32.00 \$	962.10 \$

Une fois le système démarré, ses positions, gains latents et gains totaux seront affichés dans la fenêtre ProOrder. En cliquant sur le lien "Version", vous obtiendrez une copie du code du système en question.

Un rapport détaillé des performance et une courbe de gains & pertes du système peuvent être obtenus en utilisant le bouton correspondant, comme illustré ci-dessous.

Voici un exemple d'une courbe de gains & pertes d'un système en cours d'exécution, avec le rapport détaillé de ses performances :



Rapport détaillé		ProOrder AutoTrading		EUR/USD Strategy 2.0		Spot EUR/USD	
Statistiques des positions clôturées		EUR/USD Strategy 2.0 28 juil. 2022, 17:22:28		Début : 28 juil. 2022, 17:22:41		Actuel : 4 août 2022, 11:32:00	
Gains et pertes		Tous les trades	Trades longs	Trades courts			
Gain	1 607,00 \$	1 607,00 \$	1 449,00 \$	158,00 \$			
% Gain	∞ %	∞ %	∞ %	∞ %			
Gain brut	6 150,00 \$	6 150,00 \$	5 992,00 \$	158,00 \$			
Perte brute	-4 543,00 \$	-4 543,00 \$	-4 543,00 \$	0,00 \$			
Ratio Gains/Pertes	1,35	1,35	1,32	n/d			
Gain moyen	13,50 \$	13,50 \$	12,49 \$	52,67 \$			
Moyenne des trades gagnants	57,48 \$	57,48 \$	57,62 \$	52,67 \$			
Moyenne des trades perdants	-378,58 \$	-378,58 \$	-378,58 \$	n/d			
Plus gros gain	109,00 \$	109,00 \$	109,00 \$	58,00 \$			
Plus grosse perte	-412,00 \$	-412,00 \$	-412,00 \$	n/d			
Runup max	1 619,00 \$	1 619,00 \$	1 619,00 \$	100,00 \$			
% Runup max	-411,96 %	-411,96 %	-365,46 %	-200 %			
Nb max gains consécutifs	27	27	27	3			
Drawdown max	-393,00 \$	-393,00 \$	-443,00 \$	-50,00 \$			
% Drawdown max	-∞ %	-∞ %	-∞ %	-∞ %			
Nb max pertes consécutives	2	2	2	0			
Ecart type gains & pertes	132,93 \$	132,93 \$	134,48 \$	3,77 \$			
Ratio de Sharpe	1,93	1,93	1,79	31,09			
Rendement annualisé	0 %	0 %	0 %	0 %			
AHPR	∞ %	∞ %	∞ %	37,96 %			
GHPR	∞ %	∞ %	∞ %	36,11 %			
Risk/Reward ratio	0,15	0,15	0,15	n/d			
Z-Score	0,47	0,47	0,5	0			
Frais de courtage	0,00 \$	0,00 \$	0,00 \$	0,00 \$			
Répartition des trades		Tous les trades	Trades longs	Trades courts			
Risques		Tous les trades	Trades longs	Trades courts			
Temps		Tous les trades	Trades longs	Trades courts			

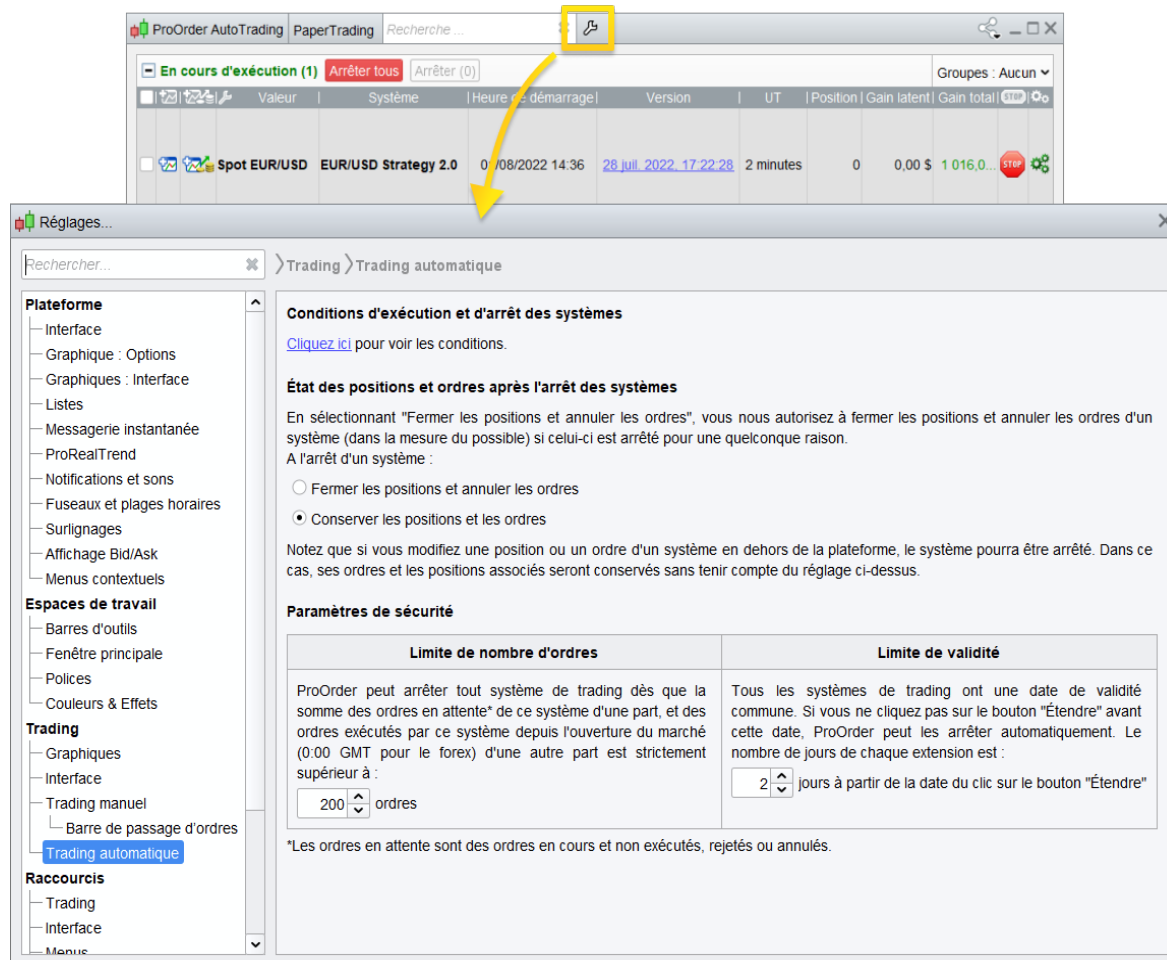
Les statistiques ci-dessus sont relatives au passé. Les performances passées ne présagent pas de l'avenir.

Note : La valeur du gain affichée dans les "Statistiques des positions clôturées" peut différer de la valeur de la courbe des gains & pertes. En effet, celle-ci tient compte des positions ouvertes en cours, ce qui n'est pas le cas du rapport détaillé.

Paramètres du Trading Automatique et conditions d'exécution

Paramètres des systèmes de trading

Il est conseillé de configurer ses préférences de trading avant de lancer un quelconque système de trading automatique. Pour cela, cliquer sur le bouton contenant la clé à molette comme illustré ci-dessous :



Un lien vers les conditions d'exécution de vos systèmes de trading est également disponible dans cette fenêtre. Il est fortement recommandé de lire ce contenu pour en apprendre plus sur le comportement de vos systèmes de trading automatique.

Arrêt automatique du système de trading

Date de validité : Tous les systèmes de trading en cours ont une date de validité commune. Si vous ne cliquez pas sur le bouton "Étendre" avant cette date, ProOrder peut interrompre les systèmes de trading correspondant. Vous pouvez vérifier la validité de vos systèmes de trading dans la fenêtre ProOrder (la date affichée prend comme référence le fuseau horaire de votre ordinateur). Pour étendre la validité de votre système, cliquez sur le bouton "Étendre" en bas de la fenêtre ProOrder pendant qu'un système de trading est en cours.

La durée d'extension peut être configurée depuis l'onglet "Trading Automatique" de la fenêtre "Préférences de Trading". Il est possible de changer cette durée même lorsque des systèmes sont en cours d'exécution : vos changements seront alors appliqués lors de la prochaine extension.

The screenshot shows the ProOrder AutoTrading interface. At the top, there are tabs for "ProOrder AutoTrading", "PaperTrading", and "Recherche ...". Below the tabs, there are two main sections:

- En cours d'exécution (3)**: This section contains a table with columns: Valeur, Heure de démarrage, UT, Position, Gain latent, Gain total, and a STOP button. The table has one row:

Valeur	Heure de démarrage	UT	Position	Gain latent	Gain total	STOP
Spot EUR/USD	02/08/2022 17:38	2 minutes	100 000	-30,00 \$	1 714,40 \$	STOP
- En attente (0)**: This section contains instructions on how to add a trading system to the "En attente" section. It includes a "Rechercher valeur" dropdown, an "Ajouter" button, and a "Préparer pour le trading automatique" button. Below this, there are "Notifications par email" (Désactivées / Activer) and a "Valide jusqu'au : 6 août 2022 16:00" label with an "Étendre" button highlighted in yellow.

At the bottom left, there is a "Connecté" status indicator, and at the bottom right, there is a link "En savoir plus".

Nombre d'ordres placés : ProOrder peut interrompre un système de trading si une quantité maximum d'ordres placés sur une journée est atteinte. Vous trouverez ce paramètre dans l'onglet "Trading Automatique" de la fenêtre "Préférences de Trading". Si le nombre d'ordres en attente cumulé au nombre de positions en cours depuis que le marché est ouvert (exemple 00:00 GMT pour le Forex) est supérieur au paramètre fixé, le système s'interrompra automatiquement. Un ordre en attente est un ordre envoyé au courtier qui n'a ni été exécuté, rejeté, ou annulé.

Cela correspondra par exemple à chaque instruction de type "Stop", "Trailing Stop" ou "Target", tant que l'ordre associé n'a ni été exécuté, rejeté, ou annulé.

Également, 3 ordres limites distincts ou 3 ordres stop distincts qui n'ont ni été exécutés, rejetés ou annulés compteront comme 3 ordres en attente (que les 3 ordres soient au même niveau de prix ou non).

Prenons par exemple un paramètre d'arrêt automatique fixé à 8 ordres par jour. Depuis l'ouverture du marché, 5 ordres ont été exécutés par un système donné. Actuellement, ce système comporte 1 position en cours et 2 ordres en attente (un ordre "Target" et un ordre "Stop"). Le système a maintenant besoin d'envoyer 1 ordre supplémentaire au marché : ce 8ème ordre ne sera pas envoyé car le niveau d'arrêt fixé est atteint ($5+2+1 = 8$). En conséquence, le système sera interrompu, en commençant par la fermeture des ordres en attente, puis par la clôture des positions actives.

Rejet d'un ordre : ProOrder peut interrompre un système de trading si un trop grand nombre d'ordres de ce système ont été rejetés. La limite est fixe et définie à 10 rejets consécutifs.

Interaction entre trading manuel et trading automatique

Lorsqu'un système de trading est en cours d'exécution sur une valeur, le passage d'ordres manuel n'y est plus disponible. Le trading manuel reste cependant possible sur les autres instruments. Un bouton spécifique apparaît alors à la place de la barre de passage d'ordres manuel sur l'instrument en question, comme illustré ci-dessous :



Il est possible de cliquer sur ce bouton pour ouvrir la fenêtre ProOrder et voir le statut des systèmes de trading en cours d'exécution.

Lancer plusieurs systèmes de trading sur un même instrument

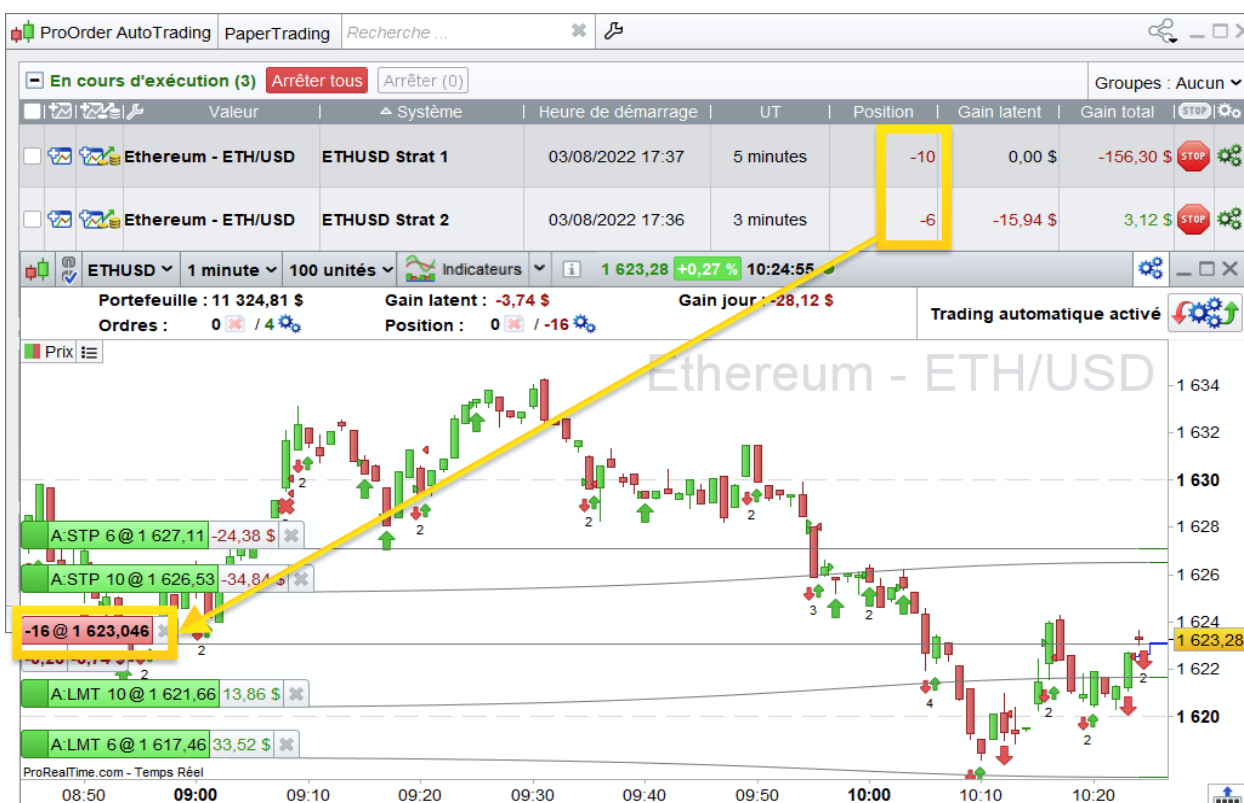
NB : ne s'applique pas à tous les brokers.

Si vous exécutez plusieurs systèmes de trading sur un même instrument, votre position nette globale est déterminée par l'ensemble de vos systèmes. Prenons un exemple où vous possédez 2 systèmes de trading. Le premier place un ordre d'achat pour 1 lot tandis que l'autre place un ordre de vente pour 1 lot. Votre position nette globale sera alors de 0. Seule la position nette globale est affichée dans le cas où plusieurs systèmes sont en cours sur un même instrument.

Lorsque que vous affichez la courbe de gains & pertes d'un système, vous obtenez également son histogramme des positions. Contrairement à la position nette globale qui cumule l'ensemble des données des systèmes de l'instrument, cet histogramme représente les positions prises par un seul système. Les niveaux de positions affichés peuvent donc être différents, comme l'illustre le graphique ci-dessous.

Exemple :

2 systèmes sont en cours d'exécution sur un même instrument : l'un est en position de vente sur 10 lots. L'autre est en position de vente sur 6 lots . La position nette est donc égale à $(10+6) = -16$.



La position nette de -16 apparaît également dans la fenêtre "Portefeuilles", disponible depuis le menu "Trading" > "Portefeuilles" :

Portefeuilles		Groupes : Aucun			Positions : 2 / Ordres : 6	Tout clôturer			
PaperTrading		Valeur : 11 324,81 \$							
Nom	%Var	Gain jour	Qté	Dernier	Gain latent	%G latent	Px.unit.	Exposition	
Ethereum - E...	+0,27 %	-28,12 \$	-16	1 623,28	-3,74 \$	-0,01 %	1 623,046 \$	25 972,48 \$	

Lorsque vous exécutez plusieurs systèmes de trading sur un même instrument, chaque système fait état de ses propres positions, ordres, trades et gains de façon indépendante. Par conséquent, les instructions de type [LongOnMarket](#) ou [ShortOnMarket](#) informent sur l'état Long ou Short du système en question.

Votre position nette globale peut différer de la position d'un système donné. La même remarque s'applique pour les variables comme [CountOfLongShares](#), [CountOfShortShares](#), [CountOfPosition](#), [PositionPrice](#), [StrategyProfit](#), [TradeIndex](#), [TradePrice](#) et [PositionPerf](#), qui ne concernent qu'un système donné.

Indicateurs non-utilisables

Les indicateurs suivants ne peuvent pas être appliqués au trading automatique car leur méthode de calcul ne permet pas leur utilisation en temps-réel :

- **ZigZag** : les signaux basés sur cet indicateur sont recalculés a posteriori. De ce fait, les signaux donnés en temps réel peuvent être très différents des signaux donnés pendant les backtests.
- **ZigZagPoint** : les signaux basés sur cet indicateur sont recalculés a posteriori. De ce fait, les signaux donnés en temps réel peuvent être très différents des signaux donnés pendant les backtests.
- **DPO**

Note sur les Fuseaux et heures de marché personnalisés

Lors de l'envoi d'un système de trading vers le module ProOrder, le fuseau horaire et les heures de marché personnalisés qui ont été définis sur le marché auquel l'instrument appartient vont être associées à la stratégie. Ces paramètres seront appliqués à chaque lancement de la stratégie.

Pour modifier le fuseau horaire ou les heures de marché personnalisées d'une stratégie, il est nécessaire de supprimer cette stratégie de ProOrder, de modifier ces paramètres dans le menu « Options/Fuseaux et plages horaires des graphiques intraday » et de renvoyer la stratégie vers ProOrder.

Vous pouvez consulter la section [Personnalisation des plages horaires](#) pour connaître les instructions concernées par le fuseau horaire du graphique ainsi que pour le fonctionnement des heures de marché personnalisées.

Annexe A : Affichage des résultats

Un système de trading présente des résultats sous 3 formes complémentaires:

Courbe de gains & pertes

La courbe de gains & pertes montre l'évolution du capital tout au long de l'exécution d'un système de trading ou d'un backtest :

- La **ligne bleue horizontale** représente le capital initial du système. Dans le cas du trading automatique, cette ligne est toujours placée à 0.

Prenons l'exemple d'un backtest avec capital initial de 10 000 et un gain de 488 : la valeur de la courbe de gains & pertes aura une valeur de 10488 comme illustré ci-dessous. Dans le cas d'un système de trading automatique avec le même gain, le point de départ de la courbe serait 0 et la valeur de fin serait de 488.



- Le **remplissage de la courbe de gains & pertes** sera de couleur bleu si la performance est positive (le point de comparaison est la ligne du capital initial). Un remplissage orange indiquera une performance négative.
- Le **trait de la courbe de gains & pertes** sera de couleur bleu pour indiquer une variation positive par rapport au niveau précédent, de couleur orange pour indiquer une variation négative.

Histogramme des positions

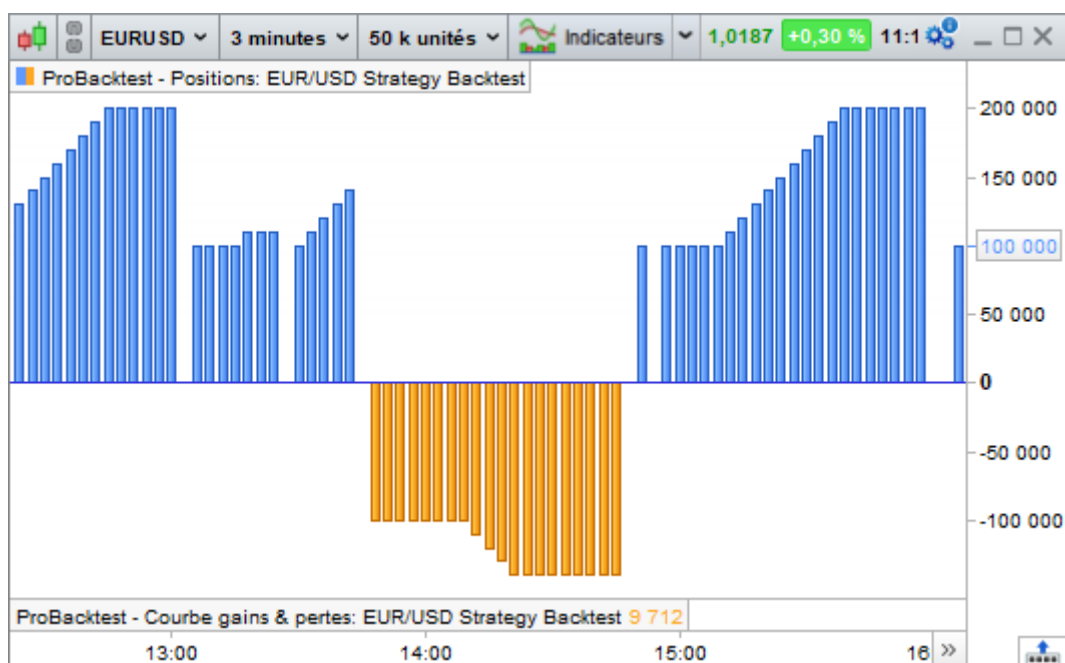
L'histogramme des positions vous permet d'afficher l'évolution des positions prises pendant l'exécution du système de trading.

- Une barre bleue indique l'ouverture d'une position longue (long).
- Une barre orange indique l'ouverture d'une position courte (short).
- Si aucune barre n'est affichée, aucune position n'est active sur le marché.

La succession de plusieurs barres de même couleur indique que la(les) position(s) est toujours en cours.

L'axe vertical à droite de l'histogramme indique la quantité des positions en cours.

Sur l'exemple ci-dessous, nous sommes en position longue d'un lot de 100 000 unités sur l'EUR/USD.



Rapport détaillé

Le rapport détaillé vous permet de visualiser les statistiques de votre système de trading ainsi que les informations de chaque position et de chaque ordre :

Rapport détaillé		ProOrder AutoTrading		EUR/USD Strategy 2.0		Spot EUR/USD	
Statistiques des positions clôturées ▼		EUR/USD Strategy 2.0 28 juil. 2022, 17:22:28		Début : 28 juil. 2022, 17:22:41		Actuel : 4 août 2022, 11:00:00	
Gains et pertes		Tous les trades	Trades longs	Trades courts			
Gain		1 607,00 \$	1 449,00 \$	158,00 \$			
% Gain		∞ %	∞ %	∞ %			
Gain brut		6 150,00 \$	5 992,00 \$	158,00 \$			
Perte brute		-4 543,00 \$	-4 543,00 \$	0,00 \$			
Ratio Gains/Pertes		1,35	1,32	n/d			
Gain moyen		13,50 \$	12,49 \$	52,67 \$			
Moyenne des trades gagnants		57,48 \$	57,62 \$	52,67 \$			
Moyenne des trades perdants		-378,58 \$	-378,58 \$	n/d			
Plus gros gain		109,00 \$	109,00 \$	58,00 \$			
Plus grosse perte		-412,00 \$	-412,00 \$	n/d			
Runup max		1 619,00 \$	1 619,00 \$	100,00 \$			
% Runup max		-411,96 %	-365,46 %	-200 %			
Nb max gains consécutifs		27	27	3			
Drawdown max		-393,00 \$	-443,00 \$	-50,00 \$			
% Drawdown max		-∞ %	-∞ %	-∞ %			
Nb max pertes consécutives		2	2	0			
Ecart type gains & pertes		132,93 \$	134,48 \$	3,77 \$			
Ratio de Sharpe		1,93	1,79	31,09			
Rendement annualisé		0 %	0 %	0 %			
AHPR		∞ %	∞ %	37,96 %			
GHPR		∞ %	∞ %	36,11 %			
Risk/Reward ratio		0,15	0,15	n/d			
Z-Score		0,47	0,5	0			
Frais de courtage		0,00 \$	0,00 \$	0,00 \$			
Répartition des trades		Tous les trades	Trades longs	Trades courts			
Nb de trades		119	116	3			
% Trades gagnants		89,92 %	89,66 %	100 %			
Trades gagnants		107	104	3			
Trades neutres		0	0	0			
Trades perdants		12	12	0			
Risques		Tous les trades	Trades longs	Trades courts			
Temps		Tous les trades	Trades longs	Trades courts			
Les statistiques ci-dessus sont relatives au passé. Les performances passées ne présagent pas de l'avenir.							

Le rapport détaillé se présente dans une fenêtre indépendante, dans laquelle vous pouvez choisir entre 5 méthodes d'affichages des données de votre stratégie :

L'affichage « Statistiques des positions clôturées » vous donnera une vision globale des performances du système de trading (gains ou pertes, nombre de positions gagnantes etc.) ainsi que des indicateurs d'évaluation du risque. Notez que seules les positions clôturées sont prises en compte.

- **"Gain"**, désigne la plus-value réalisée au cours du système de trading :

$$\text{Profit} = \text{Montant du capital final} - \text{Montant du capital initial}$$

Cette statistique permet d'évaluer de façon absolue le potentiel de gains avec le système défini (spécifique à la période d'exécution période du système).

- **Note:** les frais de courtage définis dans la section "Paramètres de courtage" sont intégrés à ce calcul.
- **"%Gain"**, représente le profit ou la perte en pourcentage :

$$\% \text{Gain} = 100 \times \text{Profit} / \text{Montant initial du capital}$$

- **"Gain brut"** : représente la somme de tous les gains et "Pertes brut" la somme de toutes les pertes. **"Ratio gains/pertes"** est le rapport de ces 2 valeurs.

- **"Nb de trades"** indique le nombre de positions qui ont été ouvertes au cours du système.

- **"% de trades gagnants"** désigne le pourcentage de positions gagnantes à l'issu du système employé. Mathématiquement, cela se définit par :

$$\% \text{ trades gagnants} = (100 \times \text{Nombre de trades positions}) / \text{Nombre de positions}$$

- **"Gain moyen"** : représente le gain moyen par position. Cette information est utile pour déterminer l'efficacité moyenne des ordres passés, notamment lorsque l'on ne souhaite pas passer beaucoup d'ordres. Elle se définit par :

$$\text{Gain moyen} = \text{Gain} / \text{Nombre de positions}$$

- **"Gain moyen des positions gagnantes"** donne le gain des positions gagnantes et **"Perte moyenne des positions perdantes"** la perte moyenne des positions perdantes.

- **"Plus gros gain"** correspond au gain maximal réalisé sur une position donnée. **"Plus grosse perte"** est la perte maximale sur une position donnée. **"Écart type gains & pertes"** est l'écart type des résultats de chaque position.

- **"Max Drawdown"** : correspond au maximum des pertes potentielles du système de trading. On définit le "drawdown" comme l'écart entre un point donné et le point le plus haut qui lui précède sur la courbe de gains & pertes :

$$DD(n) = \text{Max } t \in [0;n] P(t) - P(n)$$

Le **"Max Drawdown"** est alors le maximum de cette valeur sur l'ensemble de la période.

$$\text{MaxDD}(N) = \text{Max } n \in [0:N] (\text{Max } t \in [0;n] P(t) - P(n))$$

- **"Max Run-up"** : correspond au maximum des gains potentiels du système de trading. On définit le "run-up" comme l'écart entre un point donné et le point le plus bas qui lui précède sur la courbe de gains & pertes :

$$RU(n) = P(n) - \text{Min } t \in [0;n] P(t)$$

- Le **"Max Run-up"** est le maximum de cette valeur sur l'ensemble de la période.

$$\text{MaxRU}(N) = \text{Max } n \in [0:N] (P(n) - \text{Min } t \in [0;n] P(t))$$

Exemple :

BARINDEX	PROFIT&PERTES	DRAWDOWN	RUNUP
1	0,00	0,00	0,00
2	15,00	0,00	-15,00
3	10,00	5,00	-10,00
4	0,00	15,00	0,00
5	15,00	0,00	-15,00
6	-10,00	25,00	0,00
7	-20,00	35,00	0,00
8	-5,00	20,00	-15,00
9	-6,00	21,00	-14,00
10	20,00	0,00	-40,00
11	5,00	15,00	-25,00

Max : -35,00 40,00

"% Exposition max au risque" : L'exposition au risque est le rapport entre la perte maximale possible sur la position et le capital courant. Le % max est donc le maximum de cette valeur sur la durée du système de trading, ramené en pourcentage. Trois formules de calculs sont à distinguer selon le type d'instrument :

- Actions :

$$\% \text{Exposition max} = \text{Max positions (Quantité * prix moyen / capital)} * 100$$

- Futures :

$$\% \text{Exposition max} = \text{Max positions (Quantité * deposit / capital)} * 100$$

- Forex :

$$\% \text{Exposition max} = \text{Max positions (Quantité * prix moyen * leverage / capital)} * 100$$

où **"leverage"** est l'effet de levier, **"deposit"** est la marge

De même, **"% Exposition moy au risque"** est la moyenne du pourcentage d'exposition au risque.

"Frais de courtage" totalise les frais de courtage générés par chaque ordre (tels que définis dans les paramètres de courtage) depuis le début du système de trading.

"% de temps dans le marché" est le rapport entre les périodes pendant lesquelles des positions ont été ouvertes et la durée totale d'exécution du système de trading.

Les 4 derniers onglets vous donnent des informations sur tous les ordres passés et les positions ouvertes au cours du système de trading :

- Dans la "**Liste des ordres**" vous retrouvez les détails sur l'heure, le sens, la quantité et le prix des ordres passés. Les ordres sont affichés selon le fuseau horaire du marché de l'instrument.
- La "**Liste des positions clôturées**" vous renseigne sur les positions ouvertes et clôturées par le système de trading (positions longues ou courtes, durée exprimée en nombre de barres, performance absolue et relative de chaque position, date d'entrée et de sortie). Si une position est encore ouverte au moment de la création du rapport, elle n'apparaîtra pas dans cette liste. Dans le cadre d'un backtest, si vous souhaitez clôturer toutes les positions à la fin du backtest, définissez spécifiquement une date de fin au lieu de sélectionner "dernière date (temps réel)".
- L'affichage « **Tableau de bord** » vous donne une vue d'ensemble personnalisable des performances de votre système de trading.
- L'affichage « **Graphique avancé** » qui vous permet d'explorer des représentations graphiques sur les différents types d'indicateurs de performance de votre stratégie

Annexe B : Exemples de codes détaillés

Avertissement : Les exemples présentés dans ce manuel sont à but pédagogique. Pour votre propre trading, vous êtes entièrement libre dans le choix de vos critères. Toutes les informations présentées sont à caractère "général" et ne sauraient, en aucun cas, représenter de quelconques informations ou conseils personnalisés ou une quelconque incitation à acheter ou vendre des instruments financiers. Les performances passées ne présagent pas des futurs résultats et ne sont pas constantes dans le temps. Tout système de trading peut vous exposer à un risque de perte supérieur à votre investissement initial.

Vous pouvez consulter notre communauté ProRealTime sur le [forum ProRealCode](#) afin d'y retrouver une [documentation en ligne](#) ainsi que de nombreux exemples.

Système de trading basé sur HeikinAshi

Ce système génère un signal d'achat lorsque, en style Heiken-Ashi, une bougie verte apparaît à la suite d'une bougie rouge.

Un signal de vente à découvert est émis si une bougie rouge apparaît à la suite d'une bougie verte.

Le principe de ce backtest réside dans la reconstruction de la vue Heikin-Ashi, sur laquelle on ne peut normalement pas appliquer de systèmes. Il est donc nécessaire d'appliquer ce code sur un graphique en style chandeliers.

```
ONCE PreviousStatus = 0
IF BarIndex = 0 THEN
  XClose = TotalPrice
  XOpen = (Open + Close) / 2
ELSE
  XClose = TotalPrice
  XOpen = (XOpen[1] + Xclose[1]) / 2
ENDIF
IF XClose >= XOpen THEN
  IF PreviousStatus <> 1 THEN
    BUY 1 SHARES AT MARKET
    PreviousStatus = 1
  ENDIF
ELSE
  IF PreviousStatus <> -1 THEN
    SELLSHORT 1 SHARES AT MARKET
    PreviousStatus = -1
  ENDIF
ENDIF
```


Avertissement : Les exemples présentés dans ce manuel sont à but pédagogique. Pour votre propre trading, vous êtes entièrement libre dans le choix de vos critères. Toutes les informations présentées sont à caractère "général" et ne sauraient, en aucun cas, représenter de quelconques informations ou conseils personnalisés ou une quelconque incitation à acheter ou vendre des instruments financiers. Les performances passées ne présagent pas des futurs résultats et ne sont pas constantes dans le temps. Tout système de trading peut vous exposer à un risque de perte supérieur à votre investissement initial.

Système de trading Breakout Range System avec Stop mobile

Il s'agit d'un système de type BreakOut, où les supports et résistances sont déterminés par les plus hauts et plus bas sur les 2 premiers chandeliers de la journée.

Si le cours croise à la hausse une ligne de résistance et que la moyenne mobile 10 périodes monte, on entre en position longue.

Une limite profit à 1% ainsi qu'un stop de protection au prix du support sont définis.

La position est fermée à 17h (heure du marché) pour ne rester position d'un jour à l'autre. Un accès intraday est nécessaire pour tester ce système de trading.

```
DEFPARAM CumulateOrders = False
DEFPARAM FlatAfter = 170000

MM = Average[10](close)
MyTarget = 1

IF INTRADAYBARINDEX = 2 THEN
    MyResistance = highest[2](high)
    MySupport = lowest[2](low)
ENDIF

// Entrée en Long:
IF MM > MM[1] AND close CROSSES OVER MyResistance THEN
    BUY 1 SHARES AT MARKET
ENDIF

SELL AT MySupport STOP
SET TARGET %Profit MyTarget
```

Avertissement : Les exemples présentés dans ce manuel sont à but pédagogique. Pour votre propre trading, vous êtes entièrement libre dans le choix de vos critères. Toutes les informations présentées sont à caractère "général" et ne sauraient, en aucun cas, représenter de quelconques informations ou conseils personnalisés ou une quelconque incitation à acheter ou vendre des instruments financiers. Les performances passées ne présagent pas des futurs résultats et ne sont pas constantes dans le temps. Tout système de trading peut vous exposer à un risque de perte supérieur à votre investissement initial.

Système de trading basé sur le stochastique lissé

Ce système repose sur l'indicateur Stochastique Lissé appliqué au prix médian (indicator1) et sur sa moyenne mobile (indicator2). Lorsque l'indicateur est supérieur à sa moyenne mobile exponentielle, on achète, sinon, on vend à découvert.

On définit également un objectif à 1% de profit

```
DEFPARAM CumulateOrders = False
REM Achat
Indicator1 = SmoothedStochastic[9,9](MedianPrice)
Indicator2 = ExponentialAverage[9](Indicator1)
// Initialisation de la variable
StopLimit = 1
c1 = (Indicator1 >= Indicator2)
REM Achat
IF c1 THEN
    BUY 1 SHARES AT MARKET
ENDIF
REM Vente à découvert
IF NOT c1 THEN
    SELLSHORT 1 SHARES AT MARKET
ENDIF
SET TARGET %PROFIT StopLimit
```

Avertissement : Les exemples présentés dans ce manuel sont à but pédagogique. Pour votre propre trading, vous êtes entièrement libre dans le choix de vos critères. Toutes les informations présentées sont à caractère "général" et ne sauraient, en aucun cas, représenter de quelconques informations ou conseils personnalisés ou une quelconque incitation à acheter ou vendre des instruments financiers. Les performances passées ne présagent pas des futurs résultats et ne sont pas constantes dans le temps. Tout système de trading peut vous exposer à un risque de perte supérieur à votre investissement initial.

Swing Trading, ADX et Moyennes Mobiles

Ce backtest repose sur l'indicateur ADX et son positionnement par rapport à la valeur 30 depuis au moins les 5 derniers jours. L'objectif est de réduire les faux signaux et de minimiser les risques. Il s'agit d'un système de trading qui présente des nombreuses conditions, ce qui limite mécaniquement le nombre d'opportunités.

```

DEFPARAM CumulateOrders = False
MyADX12 = ADX[12]
ADXperiods = 5
MyMM20 = Average[20](Close)

// ACHAT
// ADX 12 est supérieur à 30 depuis au moins 5 à 10 barres.
Condition1 = LOWEST[ADXperiods + 1](MyADX12) > 30
// Si la MME 20 jours actuelle se place entre le plus haut et le plus bas de la barre en
cours ET si la MME 20 jours précédente se situe entre le plus haut et le plus bas de la
barre précédente
Condition2 = High > MyMM20 AND Low < MyMM20 AND High[1] < MyMM20[1] AND Low[1] <
MyMM20[1]
// Si le plus haut du jour casse le plus haut de la veille
Condition3 = Dhigh(0) > Dhigh(1)
IF Condition1 AND Condition2 AND Condition3 THEN
    BUY 1 SHARES AT MARKET
ENDIF
// SHORT
// ADX 12 est supérieur à 30 depuis au moins 5 à 10 barres.
Condition4 = Condition1
// Si la MME 20 jours actuelle se place entre le plus haut et le plus bas de la barre en
cours ET si la MME 20 jours précédente se situe entre le plus haut et le plus bas de la
barre précédente
Condition5 = High > MyMM20 AND Low < MyMM20 AND High[1] > MyMM20[1] AND Low[1] >
MyMM20[1]
// Si le plus bas du jour casse le plus bas de la veille
Condition6 = Dlow(0) < Dlow(1)
IF Condition4 AND Condition5 AND Condition6 THEN
    SELLSHORT 1 SHARES AT MARKET
ENDIF

```

Avertissement : Les exemples présentés dans ce manuel sont à but pédagogique. Pour votre propre trading, vous êtes entièrement libre dans le choix de vos critères. Toutes les informations présentées sont à caractère "général" et ne sauraient, en aucun cas, représenter de quelconques informations ou conseils personnalisés ou une quelconque incitation à acheter ou vendre des instruments financiers. Les performances passées ne présagent pas des futurs résultats et ne sont pas constantes dans le temps. Tout système de trading peut vous exposer à un risque de perte supérieur à votre investissement initial.

Système de trading utilisant un compteur de positions

Inverse Fisher Transform appliqué au RSI.

Ce système se base sur l'indicateur "Inverse Fisher Transform RSI" pour placer des ordres d'achat ou de vente.

Le système place un ordre d'achat quand l'Inverse Fisher Transform RSI croise à la hausse le seuil 50 et vend si l'indicateur croise à la baisse le seuil 80.

Il place un ordre de vente à découvert quand l'Inverse Fisher Transform RSI croise à la baisse le seuil 50 et rachète quand l'Inverse Fisher Transform RSI croise à la hausse le seuil 20.

Ce système de trading est à tester sur des vues 1h pour les futures ou sur des vues journalières pour les actions.

```
// Inverse Fisher Transform appliqué au RSI
// Paramètres : n = Nombre de barres pour le calcul du RSI
n = 10
Ind = RSI[n] (Close)
x = 0.1 * (Ind - 50)
y = (EXP (2 * x) - 1) / (EXP (2 * x) + 1)
z = 50 * (y + 1)
myInverseFisherTransformsRSI = z

IF (myInverseFisherTransformsRSI CROSSES OVER 50) THEN
    BUY 1 SHARES AT MARKET
ENDIF

IF (myInverseFisherTransformsRSI CROSSES UNDER 80) THEN
    SELL AT MARKET
ENDIF

IF (myInverseFisherTransformsRSI CROSSES UNDER 50) THEN
    SELLSHORT 1 SHARES AT MARKET
ENDIF

IF (myInverseFisherTransformsRSI CROSSES OVER 20) THEN
    EXITSHORT AT MARKET
ENDIF
```

Avertissement : Les exemples présentés dans ce manuel sont à but pédagogique. Pour votre propre trading, vous êtes entièrement libre dans le choix de vos critères. Toutes les informations présentées sont à caractère "général" et ne sauraient, en aucun cas, représenter de quelconques informations ou conseils personnalisés ou une quelconque incitation à acheter ou vendre des instruments financiers. Les performances passées ne présagent pas des futurs résultats et ne sont pas constantes dans le temps. Tout système de trading peut vous exposer à un risque de perte supérieur à votre investissement initial.

Système de trading utilisant TRADEINDEX – "Inside bar"

L'exemple qui suit est un système codé sur un modèle de prix souvent utilisé : l'"Inside Bar".

- On regarde d'abord si le range du n-2^{ième} chandelier est supérieur au range du n-1^{ième} chandelier. On vérifie également que ce n-1^{ième} chandelier est vert (close > open). Dans ce cas on entre long (ordre d'achat).
- On regarde d'abord si le range du n-2^{ième} chandelier est inférieur au range du n-1^{ième} chandelier. On vérifie également que ce n-1^{ième} chandelier est rouge (close < open). Dans ce cas on entre short (ordre de vente à découvert).

La sortie de position se réalise systématiquement 3 barres après l'entrée.

```
DEFPARAM CumulateOrders = False
Condition1 = (High[2] >= High[1] AND Low[2] <= Low[1])
Condition2 = (High[2] <= High[1] AND Low[2] <= Low[1])
Condition3 = (Close[1] > Open[1])
Condition4 = (Close[1] < Open[1])

IF (Condition1 AND Condition3) THEN
    BUY 1 Share AT MARKET
ENDIF

IF LONGONMARKET AND (BarIndex - TRADEINDEX) = 3 THEN
    SELL 1 share AT MARKET
ENDIF

IF (Condition2 AND Condition4) THEN
    SELLSHORT 1 share AT MARKET
ENDIF

IF SHORTONMARKET AND (BarIndex - TRADEINDEX) = 3 THEN
    EXITSHORT AT MARKET
ENDIF
```

Avertissement : Les exemples présentés dans ce manuel sont à but pédagogique. Pour votre propre trading, vous êtes entièrement libre dans le choix de vos critères. Toutes les informations présentées sont à caractère "général" et ne sauraient, en aucun cas, représenter de quelconques informations ou conseils personnalisés ou une quelconque incitation à acheter ou vendre des instruments financiers. Les performances passées ne présagent pas des futurs résultats et ne sont pas constantes dans le temps. Tout système de trading peut vous exposer à un risque de perte supérieur à votre investissement initial.

Systèmes de trading et Money Management

Les résultats d'un système peuvent être optimisés grâce au money management.

Ces stratégies de gestion du capital peuvent être mises en place sous forme de martingales, destinées à optimiser l'espérance mathématique d'un système de trading. L'espérance mathématique correspond au gain moyen ou perte moyenne sur une série de nombreux trades. Cela implique de pouvoir au préalable estimer la probabilité qu'une opération soit gagnante et le montant gagné ou perdu.

Afin de mettre en œuvre une martingale, il peut être utile de coder des ordres stops, limites profit et stops d'inactivité, ainsi que des sous-systèmes permettant de gérer dynamiquement la taille des positions.

Stops de protection & Objectifs

Pour plus d'informations sur les stops de protection et les limites profit, reportez-vous aux sections correspondantes dans ce manuel.

Sortie après inactivité

Le code ci-dessous permet d'intégrer un ordre marché d'inactivité dans votre système. Dans l'exemple suivant, les positions sont clôturées après 10 chandeliers.

```
ONCE Count = 10
// Choix du nombre de barres à compter de l'entrée en position, après lequel la position
sera
// systématiquement clôturée...
IF ONMARKET AND (BARINDEX - TRADEINDEX + 1) > Count THEN
    IF LONGONMARKET THEN
        SELL AT MARKET
    ENDIF

    IF SHORTONMARKET THEN
        EXITSHORT AT MARKET
    ENDIF
ENDIF
```

Avertissement : Les exemples présentés dans ce manuel sont à but pédagogique. Pour votre propre trading, vous êtes entièrement libre dans le choix de vos critères. Toutes les informations présentées sont à caractère "général" et ne sauraient, en aucun cas, représenter de quelconques informations ou conseils personnalisés ou une quelconque incitation à acheter ou vendre des instruments financiers. Les performances passées ne présagent pas des futurs résultats et ne sont pas constantes dans le temps. Tout système de trading peut vous exposer à un risque de perte supérieur à votre investissement initial.

Pyramidage d'une position

Le pyramidage consiste à placer plusieurs ordres successifs de même sens, en vue de renforcer la taille de la position. Le pyramidage est ensuite effectif dès lors que plusieurs ordres sont simultanément validés, à l'exemple du système suivant.

Important : Afin de rendre le cumul des positions possible, pensez à entrer la commande **"DEFPARAM CumulateOrders=True"** en début de programme.

"Countofposition" est utilisé pour limiter la taille de position maximale du système à 3.

```
DEFPARAM CumulateOrders = True
```

```
// Ce système entre long de 1 lorsque le RSI est inférieur à 30, si l'on n'est pas déjà en position.
```

```
IF NOT ONMARKET AND RSI[14](Close) < 30 THEN  
    BUY 1 SHARES AT MARKET  
ENDIF
```

```
// Si l'on a ouvert une position longue et que le cours de clôture précédent < au cours d'ouverture actuel, alors on entre long de 1 lot à chaque fois que les conditions qui précédent sont validées, dans la limite de 3 lots au total.
```

```
IF LONGONMARKET AND Open > Close[1] AND COUNTOFPOSITION < 3 THEN  
    BUY 1 SHARES AT MARKET  
ENDIF
```

```
// Lorsque le prix croise à la baisse une moyenne mobile simple, toute la position est fermée.
```

```
IF Close Crosses Under Average[14](Close) THEN  
    SELL AT MARKET  
ENDIF
```

Grâce à la maîtrise des instructions précédentes, nous pouvons maintenant nous intéresser aux martingales. Ces techniques peuvent être ajoutées à n'importe quel système.

Avertissement : Les exemples présentés dans ce manuel sont à but pédagogique. Pour votre propre trading, vous êtes entièrement libre dans le choix de vos critères. Toutes les informations présentées sont à caractère "général" et ne sauraient, en aucun cas, représenter de quelconques informations ou conseils personnalisés ou une quelconque incitation à acheter ou vendre des instruments financiers. Les performances passées ne présagent pas des futurs résultats et ne sont pas constantes dans le temps. Tout système de trading peut vous exposer à un risque de perte supérieur à votre investissement initial.

La martingale classique

La martingale classique consiste à doubler sa position lorsque l'on a fait face à une perte, de manière à se rembourser la fois suivante et à gagner une fois la position de départ récupérée. L'inconvénient majeur d'une telle gestion du capital est qu'une série de pertes consécutives rend de plus en plus difficile voire impossible de continuer à doubler sa position. En commençant avec 1000€ par exemple, si l'on perd cinq fois successives, il faudra disposer de 1000×32 soit 32000€ pour continuer le système de trading.

En conséquence, les systèmes dérivés de la martingale sont généralement plus adaptés au trading d'actions que le trading des futures ou du Forex : les mises de départ et les enjeux sont plus importants sur ces marchés.

Il est nécessaire d'intégrer le code suivant avec vos conditions d'entrée et de sortie.

```
//*****Code à insérer au début du système*****//
ONCE OrderSize = 1
ONCE ExitIndex = -2
// On commence avec une position de 1
//*****//
//*****Code à insérer juste après les instructions liquidant une position*****//
ExitIndex = BarIndex

//*****Code à insérer en fin de système*****//
IF Barindex = ExitIndex + 1 THEN
  ExitIndex = 0
  IF PositionPerf(1) < 0 THEN
    OrderSize = OrderSize * 2
    // Si le dernier trade était perdant, alors on double la taille de la position
  ELSIF PositionPerf(1) > 0 THEN
    OrderSize = 1
    // Si le dernier trade était gagnant, alors on revient à une position de taille 1
  ENDIF
ENDIF
//*****//
// La taille de la position doit être déterminée par la variable OrderSize pour l'intégrité du code
```


Avertissement : Les exemples présentés dans ce manuel sont à but pédagogique. Pour votre propre trading, vous êtes entièrement libre dans le choix de vos critères. Toutes les informations présentées sont à caractère "général" et ne sauraient, en aucun cas, représenter de quelconques informations ou conseils personnalisés ou une quelconque incitation à acheter ou vendre des instruments financiers. Les performances passées ne présagent pas des futurs résultats et ne sont pas constantes dans le temps. Tout système de trading peut vous exposer à un risque de perte supérieur à votre investissement initial.

La grande martingale

La grande martingale est semblable à la martingale classique, sauf que l'on ne double pas seulement sa position à chaque perte, on ajoute une unité supplémentaire.

Cette martingale est plus dangereuse que la martingale classique, en cas de pertes successives, mais elle permet d'augmenter significativement les gains dans le cas contraire.

Il est nécessaire d'intégrer vos conditions d'entrée et de sortie au code suivant :

```
//*****Code à insérer au début du système*****//
ONCE OrderSize = 1 // On commence avec une position de 1
ONCE ExitIndex = -2
//*****//
//*****Code à insérer juste après les instructions liquidant une position*****//
ExitIndex = BarIndex

//*****Code à insérer en fin de système*****//
IF Barindex = ExitIndex + 1 THEN
  ExitIndex = 0
  IF PositionPerf(1) < 0 THEN
    OrderSize = OrderSize * 2 + 1 // Si le dernier trade était perdant, alors on double
    la taille de la position et on ajoute une unité.
  ELSIF PositionPerf(1) >= 0 THEN
    OrderSize = 1 // Si le dernier trade était gagnant, alors on revient à une
    position de taille 1
  ENDIF
ENDIF
//*****//
// La taille de la position doit être déterminée par la variable OrderSize pour l'inté-
gralité du code
```

Avertissement : Les exemples présentés dans ce manuel sont à but pédagogique. Pour votre propre trading, vous êtes entièrement libre dans le choix de vos critères. Toutes les informations présentées sont à caractère "général" et ne sauraient, en aucun cas, représenter de quelconques informations ou conseils personnalisés ou une quelconque incitation à acheter ou vendre des instruments financiers. Les performances passées ne présagent pas des futurs résultats et ne sont pas constantes dans le temps. Tout système de trading peut vous exposer à un risque de perte supérieur à votre investissement initial.

La Piquemouche

La Piquemouche est une autre variante de la martingale classique. En cas de perte, on augmente la taille de la position par 1 si on a moins de 3 pertes consécutives. Si on a plus de 3 pertes consécutives, on double la taille de la position. Un gain réinitialise la position à 1 unité. Ce système de gestion des positions est moins dangereux que les deux précédents, car on n'augmente pas la taille de la position de manière exponentielle avant 3 pertes successives. Il est nécessaire d'intégrer vos conditions d'entrée et de sortie au code suivant :

```

//*****Code à insérer au début du système*****//
ONCE ExitIndex = -2
ONCE OrderSize = 1
// On commence avec une position de 1
ONCE BadTrades = 0
// On initialise le compteur du nombre de trades perdants successifs
//*****//
//*****Code à insérer juste après les instructions liquidant une position*****//
ExitIndex = BarIndex
//*****Code à insérer en fin de système*****//
IF Barindex = ExitIndex + 1 THEN
  ExitIndex = 0
  IF PositionPerf(1) < 0 THEN
    BadTrades = BadTrades + 1
    IF BadTrades < 3 THEN
      // Si le dernier trade était perdant et que l'on est à moins de 3 pertes
      // successives, alors on incrémente d'une unité la taille de la
      // prochaine position.
      OrderSize = OrderSize + 1
    ELSIF BadTrades MOD 3 = 0 THEN
      // Si le dernier trade était perdant et que l'on est à plus de 3 pertes
      // successives, on double la taille de la prochaine position.
      OrderSize = OrderSize * 2
    ENDIF
  ELSIF PositionPerf(1) >= 0 THEN
    // Si le dernier trade était gagnant, alors on revient à une position de taille 1.
    OrderSize = 1
    BadTrades = 0
  ENDIF
ENDIF
//*****//
// La taille de la position doit être déterminée par la variable OrderSize pour l'inté-
//gralité du code

```

Avertissement : Les exemples présentés dans ce manuel sont à but pédagogique. Pour votre propre trading, vous êtes entièrement libre dans le choix de vos critères. Toutes les informations présentées sont à caractère "général" et ne sauraient, en aucun cas, représenter de quelconques informations ou conseils personnalisés ou une quelconque incitation à acheter ou vendre des instruments financiers. Les performances passées ne présagent pas des futurs résultats et ne sont pas constantes dans le temps. Tout système de trading peut vous exposer à un risque de perte supérieur à votre investissement initial.

La pyramide d'Alembert

Cette martingale célèbre est l'œuvre d'Alembert, mathématicien français du XVIIIe siècle. En cas de perte, la position est augmentée d'une unité, et en cas de gain, elle est diminuée d'une unité.

Cette technique de gestion de la taille des positions considère que :

- des gains successifs diminuent la probabilité de gagner encore ensuite
- une perte augmente la chance de gagner ensuite

Il est nécessaire d'intégrer vos conditions d'entrée et de sortie au code suivant :

```
//*****Code à insérer au début du système*****//
ONCE OrderSize = 1
ONCE ExitIndex = -2
// On commence avec une position de 1
//*****//
//*****Code à insérer juste après les instructions liquidant une position*****//
ExitIndex = BarIndex

//*****Code à insérer en fin de système*****//
IF Barindex = ExitIndex + 1 THEN
  ExitIndex = 0
  IF PositionPerf(1) < 0 THEN
    OrderSize = OrderSize + 1
  ELSIF PositionPerf(1) >= 0 THEN
    OrderSize = MAX(OrderSize -1, 1)
  ENDIF
ENDIF
//*****//
// La taille de la position doit être déterminée par la variable OrderSize pour l'intégrité du code
```

Avertissement : Les exemples présentés dans ce manuel sont à but pédagogique. Pour votre propre trading, vous êtes entièrement libre dans le choix de vos critères. Toutes les informations présentées sont à caractère "général" et ne sauraient, en aucun cas, représenter de quelconques informations ou conseils personnalisés ou une quelconque incitation à acheter ou vendre des instruments financiers. Les performances passées ne présagent pas des futurs résultats et ne sont pas constantes dans le temps. Tout système de trading peut vous exposer à un risque de perte supérieur à votre investissement initial.

La contre d'Alembert

Il s'agit du système antagoniste à la pyramide éponyme, puis-qu'ici on diminue d'une unité la position en cas de perte et on l'augmente d'une unité en cas de gain.

Cette technique de gestion de la taille des positions considère que :

- des pertes successives augmentent la probabilité de perdre encore ensuite
- des gains successives augmentent la probabilité de gagner encore ensuite

Il est nécessaire d'intégrer vos conditions d'entrée et de sortie au code suivant :

```
//*****Code à insérer au début du système*****//
ONCE OrderSize = 1
ONCE ExitIndex = -2
// On commence avec une position de 1
//*****//
//*****Code à insérer juste après les instructions liquidant une position*****//
ExitIndex = BarIndex

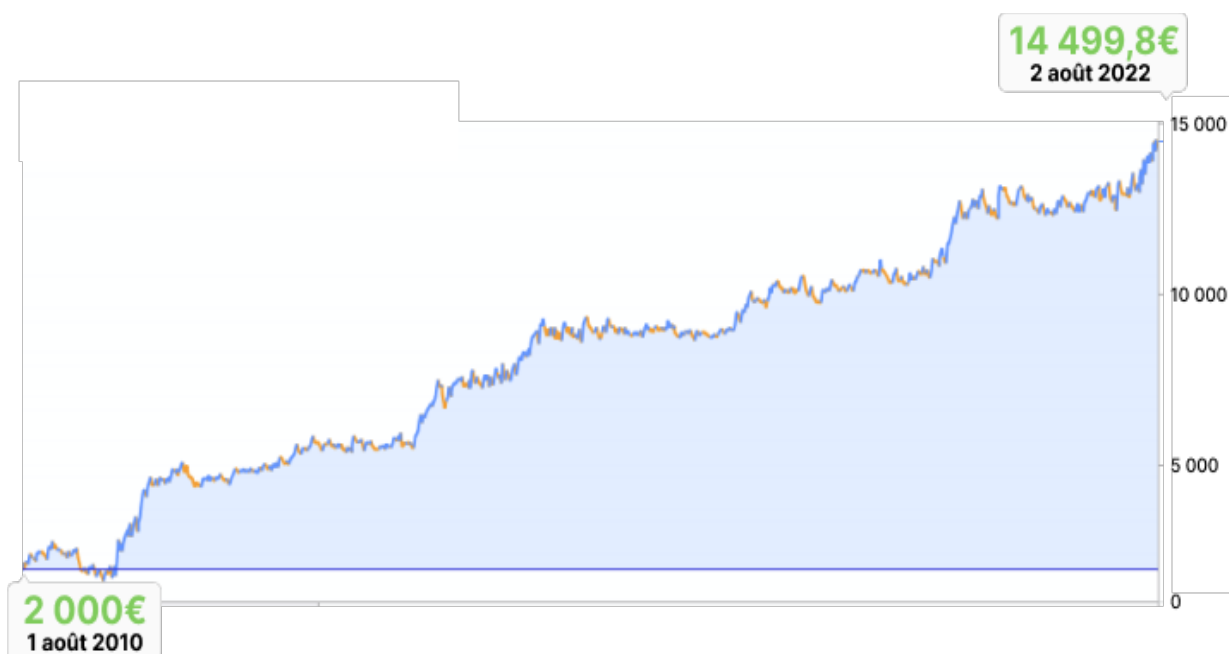
//*****Code à insérer en fin de système*****//
IF Barindex = ExitIndex + 1 THEN
  ExitIndex = 0
  IF PositionPerf(1) < 0 THEN
    OrderSize = MAX(OrderSize -1, 1)
  ELSIF PositionPerf(1) >= 0 THEN
    OrderSize = OrderSize + 1
  ENDIF
ENDIF

//*****//
// La taille de la position doit être déterminée par la variable OrderSize pour l'intégralité du code
```

Annexe C : Exemple détaillé d'un système de trading

L'exemple de système de trading décrit ci-après vous est communiqué à titre d'information uniquement. Il est destiné à vous présenter les fonctionnalités du service ProOrder. Le but de cette présentation est donc exclusivement pédagogique. En aucun cas, cette présentation, de nature simplement informative, ne saurait contenir, recommander ou suggérer, de quelque manière que ce soit, une stratégie d'investissement de la part de ProRealTime. Les performances passées ne présagent pas des futurs résultats et ne sont pas constantes dans le temps. Tout système de trading peut vous exposer à un risque de perte supérieur à votre investissement initial.

Cette annexe présente un exemple détaillé d'un système de trading de type "Breakout" (ou cassure) basé sur l'unité de temps 15 minutes et appliqué à la valeur mini-contrat CFD France 40 (1€ par point) et analyse sa performance sur ces dernières années¹ à l'aide du module de simulation ProBacktest.



Performance brute¹ du système de trading automatique "Breakout ProOrder" simulé avec ProBacktest, sur 12 ans.

Performance annualisée brute¹

16,46%

Capital de départ : **2 000,00 €** (2 août 2010)

Capital de fin : **14 499,80 €** (2 août 2022)

Gain brut¹ sur 12 ans :

**+12 499,8 €
(+ 624,9%)**

	2010	2011	2012	2013	2014	2015	2016	2017	2018	2019	2020	2021	2022
Breakout ProOrder													
Système de trading automatique	+27% ²	+76%	+7%	+12%	+33%	+21%	-0,5%	+0,9%	+9%	+6%	+21%	+3%	+10% ²
Indice CAC 40^{1,3}	-3,3% ²	-17%	+15,2%	+18%	-0,5%	+8,5%	+4,9%	+9,2%	-10,9%	+26,3%	-7,4%	+29%	-14%

¹ Les performances et gains bruts sont calculés hors frais de courtage (commissions, redevances ou autres charges), ces frais étant de nature à diminuer ces performances. ² 2010 et 2022 : années partielles. ³ Données du CAC 40 fournies par Euro-next Paris.

Présentation du système de trading automatique "Breakout ProOrder"

Un système "Breakout" classique identifie d'abord le maximum et le minimum atteints par le prix sur un laps de temps donné (dans notre exemple, les 30 premières minutes de cotation après 9h), puis place un ordre d'achat sur le niveau supérieur et un ordre de vente sur le niveau inférieur.

Le système "Breakout ProOrder" présenté ci-dessous est une version modifiée des stratégies classiques de Breakout.

Cet exemple de système "Breakout ProOrder" prend au maximum 2 positions par jour (parfois une seule voire pas du tout) entre 9h30 et 21h45. Dans tous les cas, ce système n'est plus en position après 21h45 de sorte qu'il est possible de connaître le gain ou la perte éventuelle de la journée à partir de ce moment.

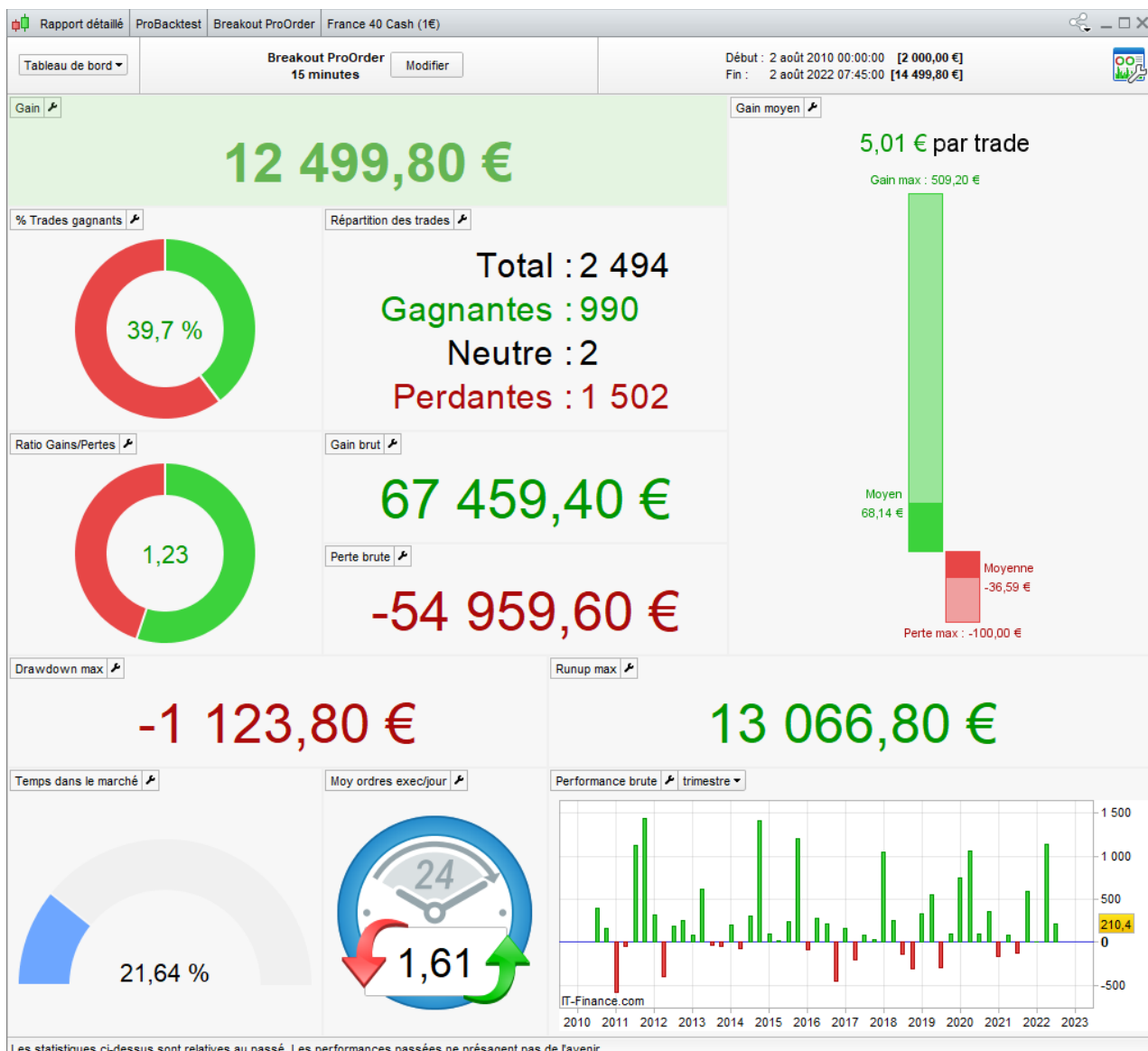
Pour aller plus loin :

- [Résultats du système de trading \(historique de 2010 à 2022\)](#)
- [Description des idées du système de trading](#)
- [Le code du système de trading](#)
- [Comment tester un code / système de trading](#)

Résultats du système de trading (historique de 2010 à 2022)

Avertissement relatif aux résultats : Les chiffres présentés ont trait aux années écoulées. Les performances passées ne sont pas un indicateur fiable des performances futures et ne sont pas constantes dans le temps.

L'image ci-dessous montre les résultats du système appliqué à la valeur CFD France 40 sur les données du passé, allant du 2 août 2010 jusqu'au 2 août 2022, soit environ 3084 jours de bourse simulés avec notre module ProBacktest.



Comme indiqué ci-dessus, sur les 2494 positions prises, la moyenne des gains en cas de position gagnante est de 68,14€ sur la période (plus gros gain de 509,20€) tandis que la moyenne des pertes en cas de position perdante est de 36,59€ sur la même période (plus grosse perte de 100€). Dans notre exemple, le nombre de positions gagnantes (990) est inférieur au nombre de positions perdantes (1502) et le montant total des gains est supérieur au montant total des pertes.

En estimant les frais provenant du spread sur le contrat CFD France 40 (1€) à 1,25€ en moyenne par ordre, les frais s'élèveraient à 6235€ sur toute la période, ce qui donnerait :

- une performance annualisée de 8,21% nette de frais avec un capital de 2 000€

A noter :

1) Dans l'exemple présenté, nous avons retenu pour cette simulation un capital initial de 2 000€, soit environ deux fois la perte maximale constatée sur toute la période depuis 2010, dans le cas où nous aurions démarré le système au moment le plus défavorable (perte maximale historique de 1123,80€). Une simulation à partir d'un montant moins élevé (par exemple, 800€) aurait également pu être envisageable dès lors que la marge actuelle demandée pour le contrat CFD France 40 (1€) est de 305€ en capital, et qu'il est pris pour hypothèse que, depuis le 1er août 2010, jour de la première prise de position dans cette simulation, le système perdrait environ 400€ et ensuite monterait régulièrement en capital.

2) Dans l'exemple présenté, la taille de position n'évolue pas alors qu'il aurait été théoriquement envisageable, que tous les 2 000€ de plus-values éventuellement obtenues, soient ajoutés 2 contrats supplémentaires à chaque nouvelle position augmentant ainsi exponentiellement la performance mais également le risque. En fait, débiter avec 2 000€ de capital et des positions de 2 contrats (même si la marge demandée par le courtier n'est que de 610€ en capital) représente un même risque de pourcentage de pertes potentielles que débiter avec 4 000€ de capital et des positions de 4 contrats par exemple.

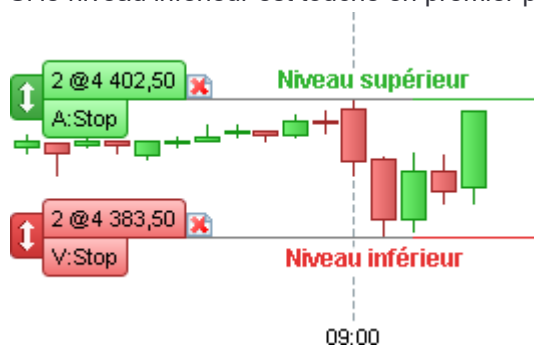
3) Un CFD est un contrat portant sur la différence de cours d'un actif entre le moment où un investisseur ouvre une position et le moment où il la clôture. Les CFD sont des produits à effet de levier. Cela signifie que l'investisseur immobilise seulement une partie de la valeur totale de son exposition sur le marché. Les CFD permettent d'augmenter significativement le rendement d'un investissement, mais les pertes potentielles peuvent également excéder le dépôt initial. Les CFD sont destinés à une clientèle avisée pouvant apprécier le niveau de risque encouru et ayant les moyens financiers de supporter un tel risque.

Description des idées du système de trading**Idee de départ : le Breakout 30 minutes**

Le Breakout classique sur 30 minutes identifie le plus haut et le plus bas atteints par la valeur du mini-contrat CFD France 40 sur les trente premières minutes de cotations significatives de la journée soit entre 9h00 et 9h30 pour cette valeur (cette période est représentée par deux chandeliers de 15 minutes chacun dans les schémas ci-dessous) et définit ces niveaux comme niveau supérieur et niveau inférieur.

Ensuite, un ordre stop d'achat est placé sur le niveau supérieur et un ordre stop de vente est placé sur le niveau inférieur comme indiqué dans le schéma de gauche ci-dessous :

- Si le niveau supérieur est touché en premier par cette valeur, une position à l'achat sera prise comme indiqué dans le schéma de droite ci-dessous.
- Si le niveau inférieur est touché en premier par cette valeur, une position à la vente sera prise.



Visualisation de l'ordre stop d'achat en vert et de l'ordre stop de vente en rouge, une fois les niveaux supérieurs et inférieurs définis.



Visualisation de la position acheteuse dans l'exemple où la valeur a d'abord atteint le niveau supérieur, ce qui ouvre une position à l'achat.

A noter :

Le système de trading n'utilise pas d'objectif de gains pour clôturer une position. Toutefois, toute position éventuellement encore ouverte à 21h45 est clôturée pour éviter le risque d'exposition la nuit.

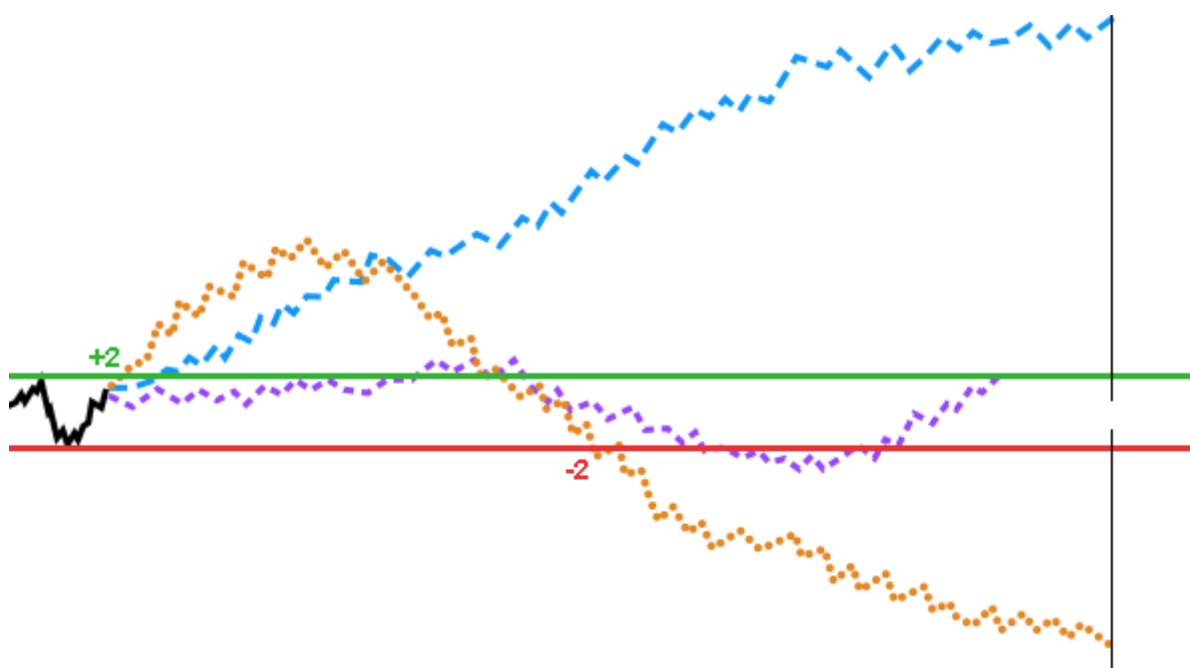
2ème idée : deux positions seront prises par jour au maximum

Dans notre exemple, nous prenons au maximum deux positions par jour (une à l'achat et une à la vente). Lorsqu'un des deux niveaux est touché en premier, une position est prise et le niveau opposé devient alors un stop de protection qui inverse la position s'il est touché par la suite.

Examinons trois des scénarios qui auraient pu se dérouler dans le cas où une position à l'achat aurait été prise en premier :

Scénario 1 :

1. Une position d'achat **+2** est prise
2. La valeur évolue toute la journée au-dessus du niveau supérieur
3. La position est clôturée à 21h45 avec un gain



Scénario 2 :

1. Une position d'achat **+2** est prise
2. La valeur baisse jusqu'au niveau inférieur.
3. Un stop de protection coupe la position avec une perte et une nouvelle position cette fois-ci de vente **-2** est prise. La position de départ est donc inversée.
4. La valeur poursuit sa baisse et cette deuxième position de vente est clôturée à 21h45 avec un gain.

Scénario 3 :

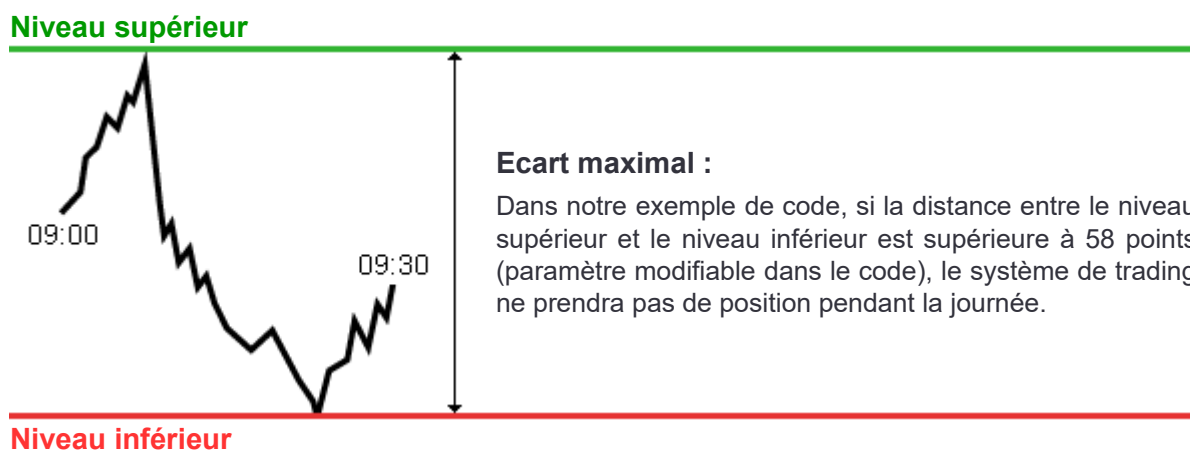
1. Une position d'achat **+2** est prise
2. La valeur baisse jusqu'au niveau inférieur.
3. Un stop de protection coupe la position avec une perte et une nouvelle position de vente **-2** est prise (position inversée).
4. La valeur remonte à nouveau jusqu'au niveau supérieur. La deuxième position vendeuse est alors clôturée par un stop de protection avec une perte (sans inverser cette fois-ci la position).

Tous les scénarios ci-dessus sont également envisageables dans le sens inverse, lorsqu'une position de vente est d'abord prise en premier.

3ème idée : limiter le risque en définissant un écart maximal entre les deux niveaux

Dans notre exemple de code, si plus de 58 points séparent le niveau supérieur du niveau inférieur, le système de trading ne prendra pas de position sur la journée.

Cette condition a été introduite pour tenter de limiter directement le risque, car comme vu dans les scénarios précédents, le système de trading peut perdre théoriquement au maximum deux fois la distance entre le niveau inférieur et le niveau supérieur.



L'écart maximal est un paramètre dans le code du système qui peut être ajusté en fonction du niveau de risque souhaité par chaque investisseur.

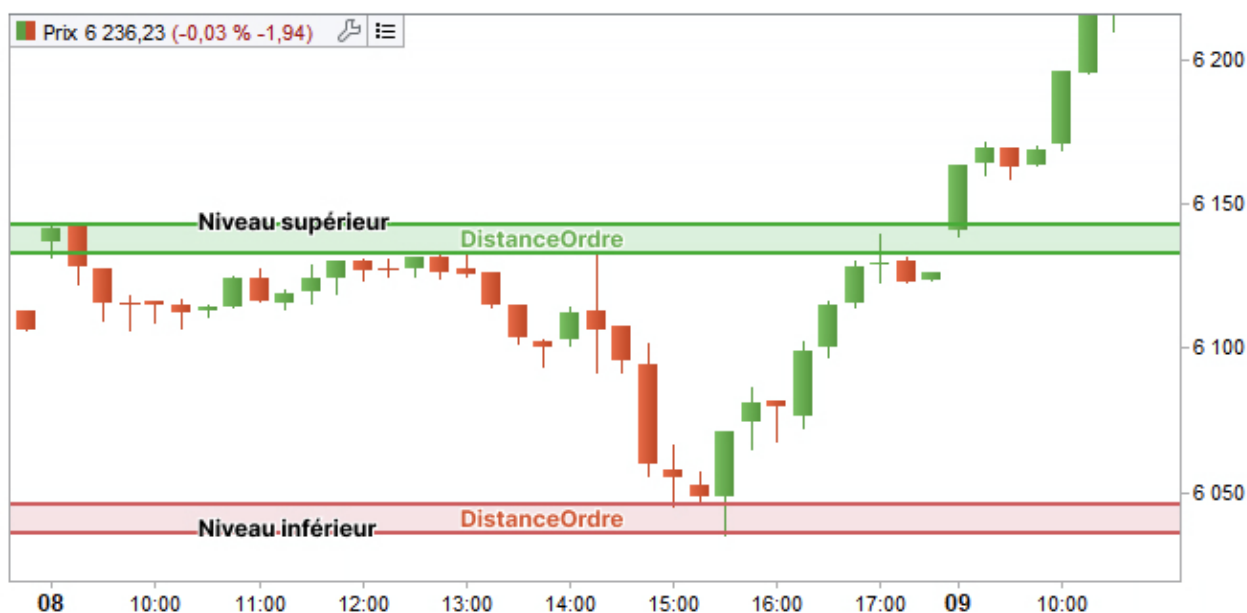
A noter :

- 1) La perte maximale théorique indiquée ci-dessus est basée sur des exécutions d'ordres aux prix des stops calculés par le système. Dans certains cas, le prix d'exécution réel peut-être différent du prix du stop demandé.
- 2) Si les bornes de cassures hautes et basses entre 9h et 9h30 sont franchies plusieurs fois par jour entre 9h30 et 21h45 (exemple du scénario 3 le plus défavorable), et ceci tous les jours à partir d'aujourd'hui, les résultats du système sont alors négatifs jour après jour sur cette période et le capital initial finirait par être perdu.
- 3) Le système de trading peut éventuellement placer une 3^{ème} position dans une même journée si l'ordre d'achat et l'ordre de vente sont tous les deux franchis dans la même période de 15 minutes après avoir défini ces deux niveaux.
- 4) Notre module de trading ProOrder permet facilement de simuler le système avec plusieurs valeurs différentes d'écart maximal. Cette optimisation de variable nous montre que la performance du système aurait pu être meilleure sur la période avec une valeur plus grande de l'écart maximal mais nous aurions alors couru un risque de perte par opération plus important car les ordres de protection auraient été plus éloignés des ordres d'entrée en position.

4ème idée : augmenter les chances d'exécution favorable

Dans notre exemple, nous sommes partis du postulat que les niveaux supérieurs et inférieurs sont des zones de support et résistance importants sur lesquels de nombreux investisseurs peuvent avoir placé des ordres. Cela se traduit souvent, lorsque le prix casse un plus haut ou un plus bas, par une accélération plus ou moins forte des prix dans le sens de la cassure sur quelques secondes.

Pour profiter de ce phénomène éventuel d'amplification de la tendance juste après la cassure et augmenter ainsi les chances que nos ordres soient favorablement exécutés, nous avons paramétré le système de manière à ce qu'il utilise uniquement des ordres stops placés à 4 points en dessous du niveau supérieur et à 4 points au-dessus du niveau inférieur, comme indiqué ci-dessous. Ce paramètre de 4 appelé "DistanceOrdre" dans le code du système peut être modifié en fonction du choix de chaque investisseur.



A noter :

1) Cette condition réduit de 2 x 4 points la distance entre l'ordre d'achat et l'ordre de vente et donc le risque maximal du système car, dans notre exemple, 50 points séparent finalement au maximum ces 2 ordres. La perte maximale théorique dans le cadre du système est donc de 200€ par jour (2 fois l'amplitude x 50€ x 2 positions).

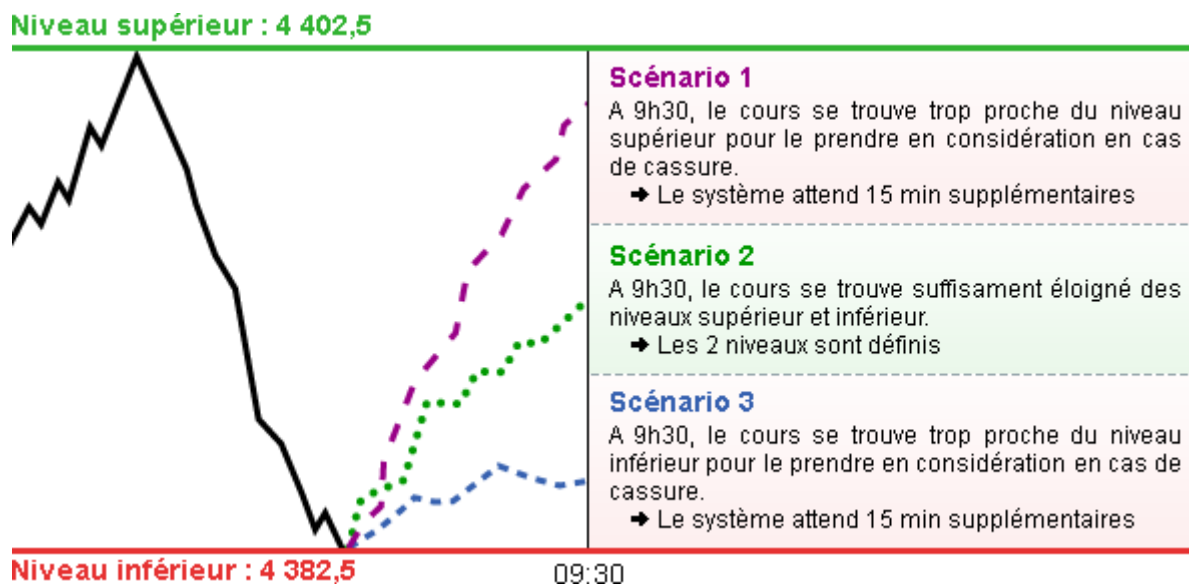
2) Dans le cas où de nombreux autres investisseurs se positionneraient aux seuils d'achat et de vente dans le même sens que le système de trading, il est possible de modifier le paramètre "DistanceOrdre" (à 4,5 ou 5 ou 5,5 ou plus) pour profiter de l'accélération qui pourrait suivre après l'exécution de son propre ordre.

5^{ème} idée : intervenir uniquement sur les breakouts (ou cassures) claires pour éviter les faux signaux

Cas 1 : valeur trop proche du niveau supérieur ou du niveau inférieur à 9h30

Dans notre exemple nous avons fait en sorte que le système ne passe pas d'ordre lorsque le cours de notre valeur de référence était trop proche du niveau supérieur ou inférieur au terme de la seconde période de 15 minutes.

Dans un tel cas de figure, le système attend le terme d'une période de 15 minutes supplémentaires.



Si à la fin de la nouvelle période de 15 minutes (soit 9h45), le cours de la valeur est toujours trop proche des niveaux, le système attend encore 15 minutes supplémentaires, et ainsi de suite jusqu'à ce que les niveaux puissent être définis. Dans tous les cas, pour un jour donné, si l'écart maximal de 58 points est dépassé, le système ne prend pas de position ce jour-là.

Ce paramètre d'éloignement des niveaux supérieur et inférieur, appelé "PourcentageMin" dans l'exemple de code du système, est ici défini à 30% mais il peut-être modifié en fonction du choix de chaque investisseur.



Dans le schéma ci-contre, le prix de clôture de chaque période est trop proche du niveau inférieur, ce jusqu'au chandelier de 10h45.

Le niveau inférieur a donc été baissé jusqu'à la clôture du chandelier de 10h45 – 11h00.

Cependant à ce moment-là, la distance entre le niveau supérieur et le niveau inférieur étant de plus de 58 points, notre exemple de système ne prendra pas de position ce jour-là.

Cas 2 : écart entre les niveaux supérieur et inférieur trop faible

Dans notre exemple, le système évite également les situations dans lesquelles l'écart entre les niveaux supérieur et inférieur est trop faible (le breakout pourrait alors être considéré comme non significatif). Le système mesure donc la distance entre l'ordre d'achat et l'ordre de vente : si cette distance est inférieure à 11 points (paramètre appelé "AmplitudeMin" et modifiable dans le code), le système ne prend pas de position sur la journée.

6^{ème} idée : ne pas prendre de position s'il ne reste pas suffisamment de temps

Etant donné que dans notre exemple de code, nous ne pouvons être en position après 21h45, nous avons paramétré l'exemple de système de telle sorte qu'il ne prenne pas de nouvelle position après 17h15.

De même, lorsqu'une journée de trading est raccourcie à cause d'un jour férié (Noël ou St-Sylvestre), le système ne prend pas de position.

Nous pouvons également personnaliser ce paramétrage dans le code de la stratégie pour modifier l'heure à partir de laquelle les signaux sont ignorés par le système.

L'exemple de système de trading décrit ci-après vous est communiqué à titre d'information uniquement. Il est destiné à vous présenter les fonctionnalités du service ProOrder. Le but de cette présentation est donc exclusivement pédagogique. En aucun cas, cette présentation, de nature simplement informative, ne saurait contenir, recommander ou suggérer, de quelque manière que ce soit, une stratégie d'investissement de la part de ProRealTime. Les performances passées ne présagent pas des futurs résultats et ne sont pas constantes dans le temps. Tout système de trading peut vous exposer à un risque de perte supérieur à votre investissement initial.

Profitez d'une assistance à la programmation [via notre formulaire dédié](#).

Code du système de trading "Breakout ProOrder"

```
// On ne charge pas de données avant le démarrage du système.
// Ainsi, le premier jour, si le système est lancée dans l'après midi,
// il attendra le lendemain pour éventuellement définir les ordres d'achat et de vente.
DEFPARAM PreLoadBars = 0
// La position est clôturée à 21h45, heure française
DEFPARAM FlatAfter = 214500
// Aucune nouvelle position prise après le chandelier qui clôture à 17h15
HeureLimite = 171500
// L'analyse du marché commence avec le chandelier 15 minutes qui clôture à 9h30
HeureDebut = 091500
// Certaines journées comme le 1er mai ou le 24 et 31 décembre sont exclues
IF (Month = 5 AND Day = 1) OR (Month = 12 AND (Day = 24 OR Day = 25 OR Day = 26 OR Day = 30 OR Day = 31)) THEN
    JourTrading = 0
ELSE
    JourTrading = 1
ENDIF

// Les variables qui peuvent être adaptées selon ses préférences
TaillePosition = 2
AmplitudeMax = 58
AmplitudeMin = 11
DistanceOrdre = 4
PourcentageMin = 30

// On initialise cette variable une seule fois au démarrage du système
ONCE DebutJourneeTrading = -1
// Les variables qui peuvent évoluer en journée sont initialisées
// à la première barre de chaque nouvelle journée de bourse
IF (Time <= HeureDebut AND DebutJourneeTrading <> 0) OR IntradayBarIndex = 0 THEN
    SeuilAchat = 0
    SeuilVente = 0
    PositionAchat = 0
    PositionVente = 0
    DebutJourneeTrading = 0
ELSIF Time >= HeureDebut AND DebutJourneeTrading = 0 AND JourTrading = 1 THEN
    // On stocke l'indice de la première barre de la journée de bourse
    IndexDebutJournee = IntradayBarIndex
    DebutJourneeTrading = 1
```

```

ELSIF DebutJourneeTrading = 1 AND Time <= HeureLimite THEN
  // Pour chaque jour de bourse, nous déterminons toutes les 15 minutes
  // la valeur la plus haute et la plus basse de l'instrument depuis HeureDebut
  // tant que les seuils d'achat et de vente ne sont pas définis
  IF SeuilAchat = 0 OR SeuilVente = 0 THEN
    NiveauSuperieur = Highest[IntradayBarIndex - IndexDebutJournee + 1](High)
    NiveauInferieur = Lowest [IntradayBarIndex - IndexDebutJournee + 1](Low)
    // Calcul de l'écart entre la valeur la plus haute
    // et la plus basse de l'instrument depuis HeureDebut
    EcartJour = NiveauSuperieur - NiveauInferieur
    // Calcul de l'écart minimal souhaité pour considérer une cassure par les prix
    // de la valeur la plus haute ou la plus basse comme une cassure significative
    EcartMin = EcartJour * PourcentageMin / 100
    // Calcul des seuils d'achat et de vente pour la journée si les conditions sont remplies
    IF EcartJour <= AmplitudeMax THEN
      IF SeuilVente = 0 AND (Close - NiveauInferieur) >= EcartMin THEN
        SeuilVente = NiveauInferieur + DistanceOrdre
      ENDIF
      IF SeuilAchat = 0 AND (NiveauSuperieur - Close) >= EcartMin THEN
        SeuilAchat = NiveauSuperieur - DistanceOrdre
      ENDIF
    ENDIF
  ENDIF
  // Création des ordres d'achat et de vente à découvert pour la journée
  // si les conditions sont remplies
  IF SeuilVente > 0 AND SeuilAchat > 0 AND (SeuilAchat - SeuilVente) >= AmplitudeMin THEN
    IF PositionAchat = 0 THEN
      IF LongOnMarket THEN
        PositionAchat = 1
      ELSE
        BUY TaillePosition CONTRACT AT SeuilAchat STOP
      ENDIF
    ENDIF
    IF PositionVente = 0 THEN
      IF ShortOnMarket THEN
        PositionVente = 1
      ELSE
        SELLSHORT TaillePosition CONTRACT AT SeuilVente STOP
      ENDIF
    ENDIF
  ENDIF
  // Définition des conditions de sortie du marché lorsqu'une position d'achat
  // ou de vente à découvert est prise
  IF LongOnMarket AND ((Time <= HeureLimite AND PositionVente = 1) OR Time > HeureLimite) THEN
    SELL AT SeuilVente STOP
  ELSIF ShortOnMarket AND ((Time <= HeureLimite AND PositionAchat = 1) OR Time > HeureLimite) THEN
    EXITSHORT AT SeuilAchat STOP
  ENDIF
  // Définition d'un risque maximal de pertes par position
  // en cas d'évolution défavorable du prix de l'instrument
  SET STOP PLOSS AmplitudeMax

```

Comment tester un code / système de trading

Mode portefeuille virtuel (PaperTrading)

Si vous avez un compte sur www.ProRealTime.com, vous pouvez exécuter vos systèmes de trading sur un portefeuille virtuel PaperTrading.

Le mode PaperTrading vous permet donc d'éprouver votre système de trading jour après jour, en conditions réelles de marché, sans risquer de capital.

Il vous permettra de voir en temps réel les prises de positions de vos systèmes de trading, et aussi de tester vos propres réactions face au trading automatique. Notez que vous pouvez réinitialiser la valeur de votre portefeuille PaperTrading autant de fois que vous le souhaitez pour recommencer une nouvelle simulation.

Mode trading réel

ProOrder AutoTrading est également disponible en mode réel via l'offre de courtage [ProRealTime IG](#) et [Pro-RealTime IB](#).

Avertissement : Si vous mettez en œuvre un système de trading au moyen du service ProOrder en mode trading réel, ce service enverra des signaux de manière automatique, selon les paramètres que vous avez déterminés, en vue de l'exécution des ordres correspondants sans qu'aucune validation individuelle de chaque ordre de votre part ne soit requise. Votre système sera exécuté automatiquement même lorsque votre ordinateur sera éteint. Il est de votre responsabilité de vous assurer que vous avez paramétré votre système de telle sorte qu'il ne conduise pas à la réalisation de pertes au-delà d'un montant que vous êtes prêt à accepter.

En tout état de cause, ProRealTime ne sera responsable des éventuelles pertes subies suite à l'exécution depuis vos systèmes de trading automatique.

Nous vous rappelons que, en raison de leur effet de levier, les CFD peuvent vous exposer à des risques de pertes supérieures à l'investissement initial. Ces produits sont destinés à une clientèle avisée pouvant apprécier le niveau de risque encouru et ayant les moyens financiers de supporter un tel risque.

Glossaire

A

CODE	IMPLÉMENTATION	FONCTION
ABS	ABS(a)	Fonction Mathématique "Valeur Absolue"
AccumDistr	AccumDistr(price)	Désigne l'Accumulation Distribution classique
ACOS	ACOS(a)	Fonction mathématique "Arc cosinus"
AdaptiveAverage	AdaptiveAverage[x,y,z](price)	Indicateur de Moyenne Adaptative
ADX	ADX[N]	Indicateur Average Directional Index
ADXR	ADXR[N]	Indicateur Average Directional Index Rate
AND	a AND b	Opérateur logique ET
ArrayMax	ArrayMax(\$MyArray)	Retourne la valeur la plus grande
ArrayMin	ArrayMin(\$MyArray)	Retourne la valeur la plus petite
ArraySort	ArraySort(\$MyArray, ASCEND) DESCEND)	Tri le tableau par ordre croissant (ASCEND) ou décroissant (DESCEND)
AroonDown	AroonDown[P]	Désigne l'Aroon Down
AroonUp	AroonUp[P]	Désigne l'Aroon Up
ATAN	ATAN(a)	Fonction mathématique "Arc tangente"
AS	RETURN x AS "ResultName"	Instruction servant à nommer une courbe
ASIN	ASIN(a)	Fonction mathématique "Arc sinus"
Average	Average[N](price)	Moyenne Mobile Arithmétique
AverageTrueRange	AverageTrueRange[N](price)	Désigne la moyenne mobile par lissage de Wilder du True Range

B

CODE	IMPLÉMENTATION	FONCTION
BarsSince	BarsSince(condition)	Retourne le nombre de chandelier depuis que la dernière condition a été remplie
BarIndex	BarIndex	Nombre de barres depuis la première barre de données chargée (dans un graphique dans le cas d'un indicateur ProBuilder ou pour un système de trading dans le cas d'un ProBacktest ou ProInvest)
BollingerBandWidth	BollingerBandWidth[N](price)	Bande passante de Bollinger
BollingerDown	BollingerDown[N](price)	Support de la bande de Bollinger
BollingerUp	BollingerUp[N](price)	Résistance de la bande de Bollinger
BREAK	(FOR...DO...BREAK...NEXT)	Instruction de sortie forcée de boucle FOR ou

CODE	IMPLÉMENTATION	FONCTION
	ou (WHILE...DO...BREAK...WEND)	WHILE
BUY	BUY (n) SHARES	Instruction d'ouverture de position longue
BREAKEVEN	SET STOP BREAKEVEN	Permet de poser un stop de protection au prix d'entrée de la position en cours.

C

CODE	IMPLÉMENTATION	FONCTION
CALCULATEONLASTBARS	DEFPARAM CalculateOnLastBars = 200	Permet d'augmenter la vitesse à laquelle un indicateur sera calculé en définissant le nombre de barres présentant le résultat.
CALL	myResult = CALL myFunction	Appel de fonction utilisateur
CASH	BUY x CASH	Désigne le montant à utiliser
CCI	CCI[N](price) ou CCI[N]	Donne le Commodity Channel Index
CEIL	CEIL(N)	Renvoie le plus petit nombre entier supérieur à N.
ChaikinOsc	ChaikinOsc[Ch1, Ch2](price)	Désigne l'oscillateur de Chaikin
Chandle	Chandle[N](price)	Désigne le Chande Momentum Oscillator
ChandeKrollStopUp	ChandeKrollStopUp[Pp, Qq, X]	Stop de protection selon Chande et Kroll en position acheteuse
ChandeKrollStopDown	ChandeKrollStopDown[Pp, Qq, X]	Stop de protection selon Chande et Kroll en position vendeuse
Close	Close[N]	Désigne le prix de clôture de la barre courante ou celui du n-ième chandelier précédent
COLOURED	RETURN x COLOURED(R,G,B)	Colorie une courbe d'une certaine couleur selon la convention RGB
COS	COS(a)	Fonction cosinus
COUNTOFLONGSHARES	COUNTOFLONGSHARES	Comptabilise le nombre de titres en position longue
COUNTOFPOSITION	COUNTOFPOSITION	Comptabilise le nombre de titres en position (soit acheteuse ou vendeuse)
COUNTOFSHORTSHARES	COUNTOFSHORTSHARES	Comptabilise le nombre de titres en position courte
CONTRACT	BUY 1 CONTRACT	Désigne le nombre de contrats à acheter. Equivalent de "Shares"
CROSSES OVER	a CROSSES OVER b	Opérateur booléen vérifiant qu'une courbe passe au-dessus d'une autre
CROSSES UNDER	a CROSSES UNDER b	Opérateur booléen vérifiant qu'une courbe passe en dessous d'une autre

<code>cumsum</code>	<code>cumsum(price)</code>	Sommation d'un prix depuis le début de l'historique affiché
<code>CurrentDayOfWeek</code>	<code>CurrentDayOfWeek</code>	Désigne le jour actuel
<code>CurrentHour</code>	<code>CurrentHour</code>	Désigne l'heure actuelle
<code>CurrentMinute</code>	<code>CurrentMinute</code>	Désigne la minute actuelle
<code>CurrentMonth</code>	<code>CurrentMonth</code>	Désigne le mois actuel
<code>CurrentSecond</code>	<code>CurrentSecond</code>	Désigne la seconde actuelle
<code>CurrentTime</code>	<code>CurrentTime</code>	Désigne HeureMinute actuelle
<code>CurrentYear</code>	<code>CurrentYear</code>	Désigne l'année actuelle
<code>CustomClose</code>	<code>CustomClose[N]</code>	Constante paramétrable dans la fenêtre de propriétés
<code>Cycle</code>	<code>Cycle(price)</code>	Indicateur Cycle

D

CODE	IMPLÉMENTATION	FONCTION
<code>Date</code>	<code>Date[N]</code>	Désigne la date de clôture de la barre courante
<code>DATETOBARINDEX</code>	<code>DATETOBARINDEX(date)</code>	Permet d'utiliser une date pour les fonctions de dessins.
<code>Day</code>	<code>Day[N]</code>	Jour de clôture de la barre courante
<code>Days</code>	<code>Days[N]</code>	Compteur de jours depuis 1900
<code>Days</code>	<code>TIMEFRAME(X Days)</code>	Définit la période à « X Jours » pour la suite des calculs du code.
<code>DayOfWeek</code>	<code>DayOfWeek[N]</code>	Désigne le jour de la semaine durant lequel la barre courante a clos
<code>DClose</code>	<code>DClose(N)</code>	Prix de clôture de la n-ième journée antérieure à celle de la barre courante
<code>DEFPARAM</code>	<code>DEFPARAM</code>	Permet de définir des paramètres
<code>DEMA</code>	<code>DEMA[N](price)</code>	Double Moyenne Mobile Exponentielle
<code>DHigh</code>	<code>DHigh(N)</code>	Prix le plus haut de la n-ième journée antérieure à celle de la barre courante
<code>DI</code>	<code>DI[N](price)</code>	Désigne le Demand Index
<code>DIminus</code>	<code>DIminus[N](price)</code>	Désigne le DI-
<code>DIplus</code>	<code>DIplus[N](price)</code>	Désigne le DI+
<code>DivergenceCCI</code>	<code>DivergenceCCI[Div1,Div2,Div3,Div4]</code>	Indicateur de détection de divergences entre le prix et le CCI.
<code>DivergenceMACD</code>	<code>DivergenceMACD[Div1,Div2,Div3,Div4](close)</code>	Indicateur de détection de divergences entre le prix et le MACD.
<code>DLow</code>	<code>DLow(N)</code>	Prix le plus bas de la n-ième journée antérieure à celle de la barre courante

CODE	IMPLÉMENTATION	FONCTION
DO	Voir FOR et WHILE	Instruction facultative des FOR et WHILE pour l'action de bouclage
DonchianChannelCenter	DonchianChannelCenter[N]	Canal du milieu de l'indicateur Donchian pour N périodes.
DonchianChannelDown	DonchianChannelDown[N]	Canal inférieur de l'indicateur Donchian pour N périodes.
DonchianChannelUP	DonchianChannelUp[N]	Canal supérieur de l'indicateur Donchian pour N périodes.
DOpen	DOpen(N)	Prix d'ouverture de la n-ième journée antérieure à celle de la barre courante
DOWNTO	Voir FOR	Instruction sur boucle FOR pour une lecture décroissante
DPO	DPO[N](price)	Désigne le Detrented Price Oscillator
DynamicZoneRSIDown	DynamicZoneRSIDown[rsiN, N]	Bande inférieure de l'indicateur Dynamic Zone RSI.
DynamicZoneRSIUp	DynamicZoneRSIUp[rsiN, N]	Bande supérieure de l'indicateur Dynamic Zone RSI.
DynamicZoneStochasticDown	DynamicZoneStochasticDown[N]	Bande inférieure de l'indicateur Dynamic Zone Stochastic.
DynamicZoneStochasticUp	DynamicZoneStochasticUp[N]	Bande supérieure de l'indicateur Dynamic Zone Stochastic.

E

CODE	IMPLÉMENTATION	FONCTION
EaseOfMovement	EaseOfMovement[I]	Désigne l'indicateur Ease of Movement
ElderrayBearPower	ElderrayBearPower[N](close)	Désigne l'indicateur Elderray Bear Power
ElderrayBullPower	ElderrayBullPower[N](close)	Désigne l'indicateur Elderray Bull Power
ELSE	Voir IF/THEN/ELSE/ENDIF	Instruction d'appel de la seconde condition à défaut de la première issue du IF
ELSIF	Voir IF/THEN/ELSE/ENDIF	Contraction de ELSE IF
EMV	EMV[N]	Désigne l'indicateur Ease of Movement Value
ENDIF	Voir IF/THEN/ELSE/ENDIF	Instruction de clôture des instructions conditionnelles
EndPointAverage	EndPointAverage[N](price)	Moyenne Mobile à dernier point
EXITSHORT	EXITSHORT x SHARES	Instruction qui clôture une position courte
EXP	EXP(a)	Fonction Mathématique "Exponentielle"
ExponentialAverage	ExponentialAverage[N](price)	Moyenne Mobile Exponentielle

F - G

CODE	IMPLÉMENTATION	FONCTION
FractalDimensionIndex	FractalDimensionIndex[N] (close)	Désigne l'indicateur Fractal Dimension Index.
FOR/TO/NEXT	FOR i =a TO b DO a NEXT	Prend les valeurs désignées du début à la fin ou vice versa
FLATAFTER	DefParam FlatAfter = HHMM-SS	Annule n'importe quel ordre en attente, ferme n'importe quelle position ouverte et empêche l'ajout éventuel d'ordres additionnels après l'heure paramétrée (en heures, minutes et secondes) dans le fuseau horaire utilisateur
FLATBEFORE	Defparam FlatBefore = HHMMSS	Annule n'importe quel ordre en attente, ferme n'importe quelle position ouverte et empêche l'ajout éventuel d'ordres additionnels avant l'heure paramétrée (en heures, minutes et secondes) dans le fuseau horaire utilisateur
ForceIndex	ForceIndex(price)	Indicateur Force Index déterminant qui contrôle le marché
FLOOR	FLOOR(N)	Renvoie le plus grand nombre entier inférieur à N.
GRAPH	GRAPH myvariable AS "myvariable"	Instruction backtest qui permet de visualiser les valeurs des variables sur les données historiques
GRAPHONPRICE	GRAPHONPRICE myvariable AS "myvariable"	Instruction backtest qui permet de visualiser les valeurs des variables sur le graphique du prix.

H

CODE	IMPLÉMENTATION	FONCTION
High	High[N]	Désigne le plus haut de la barre courante ou celle de n barres auparavant
Highest	highest[N](price)	Désigne le plus haut cours sur un horizon donné
HistoricVolatility	HistoricVolatility[N](price)	Désigne la volatilité historique ou statistique
Hour	Hour[N]	Désigne l'heure de clôture de chaque barre dans le fuseau horaire utilisateur

I - J - K

CODE	IMPLÉMENTATION	FONCTION
IF/THEN/ENDIF	IF a THEN b ENDIF	Ensemble d'instructions conditionnelles sans deuxième condition
IF/THEN/ELSE/ENDIF	IF a THEN b ELSE c ENDIF	Ensemble d'instructions conditionnelles
IntradayBarIndex	IntradayBarIndex[N]	Compte le nombre de chandeliers sur le gra-

CODE	IMPLÉMENTATION	FONCTION
		phique intraday
<code>IsSet</code>	<code>IsSet(\$MyArray[index])</code>	Retourne 1 si l'indice (index) du tableau a été défini, 0 si il n'a pas été défini.
<code>KeltnerBandCenter</code>	<code>KeltnerBandCenter[N]</code>	Bande centrale de l'indicateur Keltner pour N période.
<code>KeltnerBandDown</code>	<code>KeltnerBandDown[N]</code>	Bande inférieure de l'indicateur Keltner pour N période.
<code>KeltnerBandUp</code>	<code>KeltnerBandUp[N]</code>	Bande supérieure de l'indicateur Keltner pour N période.
<code>KijunSen</code>	<code>KijunSen[9,26,52]</code>	Retourne la valeur KijunSen de l'indicateur Ichimoku

L

CODE	IMPLÉMENTATION	FONCTION
<code>LastSet</code>	<code>LastSet(\$MyArray)</code>	Retourne l'index défini le plus élevé du tableau
<code>LIMIT</code>	<code>BUY AT x LIMIT</code>	Instruction qui introduit un ordre Limite
<code>LinearRegression</code>	<code>LinearRegression[N](price)</code>	Droite de régression linéaire
<code>LinearRegressionSlope</code>	<code>LinearRegressionSlope[N](price)</code>	Pente de la droite de régression linéaire
<code>LOG</code>	<code>LOG(a)</code>	Fonction mathématique "logarithme népérien"
<code>LONGONMARKET</code>	<code>LONGONMARKET</code>	Indique si vous avez des positions acheteuses (=longues) sur le marché
<code>LONGTRIGGERED</code>	<code>LONGTRIGGERED[N]</code>	Indique si une ouverture de position longue a eu lieu sur le chandelier N
<code>Low</code>	<code>Low[N]</code>	Désigne le plus bas atteint durant la période
<code>lowest</code>	<code>lowest[N](price)</code>	Désigne le plus bas d'une période sur un horizon donné
<code>LOSS</code>	<code>SET STOP LOSS x</code>	Permet de poser un stop loss à x unités du cours d'entrée en position
<code>%LOSS</code>	<code>SET STOP %LOSS x</code>	Définit une perte à x % du cours d'entrée en position
<code>\$LOSS</code>	<code>SET STOP \$LOSS x</code>	Définit une perte de x €, \$ (devise de l'instrument)
<code>PLOSS</code>	<code>SET STOP PLOSS x</code>	Définit une perte à x point du cours d'entrée en position
<code>LOT</code>	<code>BUY 1 LOT</code>	Désigne le nombre de lot à acheter (équivalent de "SHARE")

M

CODE	IMPLÉMENTATION	FONCTION
MACD	MACD[S,L,Si](price)	Moving Average Convergence Divergence (MACD)
MACDline	MACDLine[S,L,Si](price)	Désigne la ligne du MACD
MACDSignal	MACDSignal[S,L,Si](price)	Désigne le Signal du MACD.
MARKET	BUY AT MARKET	Désigne un ordre au prix de marché. il sera exécuté à l'ouverture de la barre suivante
MassIndex	MassIndex[N]	Indicateur Mass Index appliqué sur N barres
MAX	MAX(a,b)	Fonction mathématique "Maximum"
MedianPrice	MedianPrice	Moyenne du prix le plus haut et du plus bas
MIN	MIN(a,b)	Fonction Mathématique "Minimum"
Minute	Minute	Désigne la minute du moment de la clôture de chaque barre de l'historique
Minutes	TIMEFRAME(X Minutes)	Définit la période à « X Minutes » pour la suite des calculs du code.
MOD	a MOD b	Fonction Mathématique "Reste de la division euclidienne"
Momentum	Momentum[I]	Momentum (prix de clôture – prix de clôture de la n-ième barre précédente)
MoneyFlow	MoneyFlow[N](price)	Donne le MoneyFlow entre -1 et 1
MoneyFlowIndex	MoneyFlowIndex[N]	Désigne le MoneyFlowIndex
Month	Month[N]	Désigne le mois de la clôture de chaque barre de l'historique
Months	TIMEFRAME(X Months)	Définit la période à « X Mois » pour la suite des calculs du code.

N

CODE	IMPLÉMENTATION	FONCTION
NegativeVolumeIndex	NegativeVolumeIndex[N]	Désigne l'indice de volume négatif
NEXT	Voir FOR/TO/NEXT	Instruction à placer à la fin de la boucle "FOR"
NextBarOpen	AT MARKET NextBarOpen	Désigne un ordre exécuté à l'ouverture de la barre suivante.
NOCASHUPDATE	DEFPARAM NOCASHUPDATE=true/false	Permet de ne pas actualiser le capital avec les gains/pertes (backtests seulement)
NOT	Not A	Opérateur logique NON

O

CODE	IMPLÉMENTATION	FONCTION
OBV	OBV(price)	Désigne l' "On-Balance-Volume"
ONCE	ONCE VariableName = VariableValue	Instruction qui en précède une autre qu'on ne veut réaliser qu'une seule fois
ONMARKET	ONMARKET	Indique si vous êtes en position
Open	Open[N]	Désigne le prix d'ouverture de la barre courante ou celui du n-ième chandelier précédent
OpenDay	OpenDay[N]	Désigne le jour d'ouverture de la barre courante ou celui du n-ième chandelier précédent
OpenDayOfWeek	OpenDay[N]	Désigne le jour de la semaine de l'ouverture de la barre courante ou celui du n-ième chandelier précédent
OpenHour	OpenHour[N]	Désigne l'heure d'ouverture de la barre courante ou celui du n-ième chandelier précédent
OpenMinute	OpenMinute[N]	Désigne la minute d'ouverture de la barre courante ou celui du n-ième chandelier précédent
OpenMonth	OpenMonth[N]	Désigne le mois d'ouverture de la barre courante ou celui du n-ième chandelier précédent
OpenSecond	OpenSecond[N]	Désigne la seconde d'ouverture de la barre courante ou celui du n-ième chandelier précédent
OpenTime	OpenTime[N]	Désigne le temps (HHMMSS) d'ouverture de la barre courante ou celui du n-ième chandelier précédent
OpenTimestamp	OpenTime[N]	Désigne le timestamp UNIX d'ouverture de la barre courante ou celui du n-ième chandelier précédent
OpenWeek	OpenWeek[N]	Désigne la semaine d'ouverture de la barre courante ou celui du n-ième chandelier précédent
OpenYear	OpenYear[N]	Désigne l'année d'ouverture de la barre courante ou celui du n-ième chandelier précédent
OR	a OR b	Opérateur logique OU

P - Q

CODE	IMPLÉMENTATION	FONCTION
PIPVVALUE	PipValue	Valeur en €/ \$ d'un pip (ou point), PipValue=Pointvalue
Pipsize	Pipsize	Taille d'un pip (forex)
POINTVALUE	PointValue	Valeur en €/ \$ d'un pip (ou point), PipValue=Pointvalue
POINTSIZ	PointSize	Taille d'un pip (ou point) : PipSize=PointSize

CODE	IMPLÉMENTATION	FONCTION
POSITIONPERF	PositionPerf(n)	Indique le pourcentage de gain ou de perte de la n-ième position cloturée
POSITIONPRICE	PositionPrice	Indique le prix moyen de la position en cours
PRELOADBARS	DEFPARAM PRELOADBARS = 200	Indique le montant maximum de barres préchargées pour le calcul d'indicateurs utilisés dans un système de trading.
PROFIT	SET TARGET PROFIT x	Permet de poser une target profit à x unités du cours d'entrée en position
%PROFIT	SET TARGET %PROFIT x	Définit un gain à x % du cours d'entrée en position
\$PROFIT	SET TARGET \$PROFIT x	Définit un gain de x €, \$ (devise de l'instrument)
PPROFIT	SET TARGET PPROFIT x	Définit un gain à x point du cours d'entrée en position
PositiveVolumeIndex	PositiveVolumeIndex(price)	Désigne l'indicateur Positive Volume Index
POW	POW(N,P)	Retourne la valeur de N à la puissance P.
PriceOscillator	PriceOscillator[S,L](price)	Indicateur Pourcentage Price oscillator
PRTBANDSUP	pbUp = PRTBANDSUP	Donne la valeur de la bande supérieure de PRTBands
PRTBANDSDOWN	pbDown = PRTBANDSDOWN	Donne la valeur de la bande inférieure de PRTBands
PRTBANDSHORTTERM	pbShort = PRTBANDSSHORTTERM	Donne la valeur de la bande court terme de PRTBands
PRTBANDMEDIUMTERM	pbMedium = PRTBANDSMEDIUMTERM	Donne la valeur de la bande long terme de PRTBands
PVT	PVT(price)	Désigne l'indicateur "Price Volume Trend"
QUIT	QUIT	Instruction pour stopper un système de trading

R

CODE	IMPLÉMENTATION	FONCTION
R2	R2[N](price)	Coefficient R Carré (taux d'erreur des prix à la regression linéaire)
RANDOM	RANDOM(Min, Max)	Génère un nombre entier aléatoire entre les bornes Min et Max incluses.
Range	Range[N]	Différence entre le prix le plus haut et le plus bas de la barre courante
Repulse	Repulse[N](price)	Mesure la poussée haussière et baissière de chaque bougie
RepulseMM	RepulseMM[N,PeriodMM,factorMM](price)	Ligne MM de l'indicateur Repulse.
ROC	ROC[N](price)	Désigne le "Price Rate of Change"

RocnRoll	RocnRoll(price)	Désigne l'indicateur RocnRoll basé sur l'indicateur ROC.
ROUND	ROUND(a)	Fonction mathématique "Arrondi à l'unité"
RSI	RSI[N](price)	Désigne l'oscillateur "Relative Strength Index"

S

CODE	IMPLÉMENTATION	FONCTION
SansSerif	DRAWTEXT(« text »,barindex,close,SansSerif,Italic, 10)	Police SansSerif appliquée au texte
SAR	SAR[At,St,Lim]	Désigne le Parabolique SAR
SELL	SELL (n) SHARES	Instruction de clôture de position longue
SELLSHORT	SELLSHORT (n) SHARES	Instruction d'ouverture de position courte
SET	SET	Permet de poser un stop ou une limite
SHARES	BUY (n) SHARES	Désigne le nombre d'actions à acheter
SHORTONMARKET	SHORTONMARKET	Indique si vous avez des positions vendeuses (=courtes) sur le marché
SHORTTRIGGERED	SHORTTRIGGERED[N]	Indique si une ouverture de position courte a eu lieu sur le chandelier N
SARatdmf	SARatdmf[At,St,Lim](price)	Parreil (parabolique SAR) Désigne le Parabolique SAR ATDMF
Seconds	TIMEFRAME(X Seconds)	Définit la période à « X Secondes » pour la suite des calculs du code.
SenkouSpanA	SenkouSpanA[9,26,52]	Retourne la valeur SenkouSpanA de l'indicateur Ichimoku
SenkouSpanB	SenkouSpanB[9,26,52]	Retourne la valeur SenkouSpanB de l'indicateur Ichimoku
Serif	DRAWTEXT(« text »,barindex,close,Serif,Italic, 10)	Police Serif appliquée au texte
SIN	SIN(a)	Fonction Mathématique "Sinus"
SGN	SGN(a)	Fonction Mathématique "Signe de"
SMI	SMI[N,SS,DS](price)	Désigne le Stochastic Momentum Index
SmoothedRepulse	SmoothedRepulse[N](price)	Désigne un Repulse lissée
SmoothedStochastic	SmoothedStochastic[N,K](price)	Désigne une Stochastique lissée
SQUARE	SQUARE(a)	Fonction mathématique "Mise au carré"
SQRT	SQRT(a)	Fonction Mathématique "Mise à la racine carrée"
Standard	DRAWTEXT(« text »,barindex,close,Serif,Standard, 10)	Style standard appliqué au texte

CODE	IMPLÉMENTATION	FONCTION
STD	STD[N](price)	Fonction Statistique "écart-type"
STE	STE[N](price)	Fonction Statistique "écart-erreur"
STYLE	STYLE(dottedline, width)	Applique le type de style <i>dottedline</i> avec une épaisseur <i>width</i> sur un objet.
Stochastic	Stochastic[N,K](price)	Ligne %K de la Stochastique
Stochasticd	Stochasticd[N,K,D](price)	Ligne %D de la Stochastique
STOP	SET STOP LOSS	Permet de poser un stop (cf glossaire LOSS)
summation	summation[N](price)	Somme d'un certain prix des N derniers chandeliers
Supertrend	Supertrend[STF,N]	Désigne le Super Trend

T

CODE	IMPLÉMENTATION	FONCTION
TAN	TAN(a)	Fonction mathématique "Tangente"
TEMA	TEMA[N](price)	Moyenne Mobile Exponentielle Triple
TARGET	SET TARGET PROFIT x	Permet de poser un objectif à x unités du cours
TenkanSen	TenkanSen[9,26,52]	Retourne la valeur TenkanSen de l'indicateur Ichimoku
THEN	Voir IF/THEN/ELSE/ENDIF	Instruction suivant la première condition de l'instruction "IF"
Ticks	TIMEFRAME(X Ticks)	Définit la période à « X Ticks » pour la suite des calculs du code.
TicksSize	TicksSize	Variation minimale du cours
Time	Time[N]	Donne l'HeureMinuteSeconde et permet de faire appel à l'heure
TimeSeriesAverage	TimeSeriesAverage[N](price)	Moyenne mobile des séries temporelles
Timestamp	Timestamp[N]	Date UNIX de la clôture du chandelier N.
TO	Voir FOR/TO/NEXT	Instruction "jusqu'à" dans la boucle "Pour"
Today	YYYYMMDD	Date de la journée actuelle
TotalPrice	TotalPrice[N]	(Clôture + Ouverture + Plus Haut + Plus Bas) / 4
TR	TR(price)	Désigne le True Range
TRADEINDEX	TRADEINDEX(n)	Indique l'indice de la barre sur laquelle a été exécuté le n-ième dernier ordre
TRADEPRICE	TRADEPRICE(n)	Indique le niveau de cours du n-ième dernier ordre exécuté
TRAILING	SET STOP TRAILING x	Permet de poser un stop suiveur à x unités du cours

CODE	IMPLÉMENTATION	FONCTION
<code>%TRAILING</code>	<code>SET STOP %TRAILING x</code>	Place un trailing stop à x % du cours
<code>\$TRAILING</code>	<code>SET STOP \$TRAILING x</code>	Place un trailing stop tel que la perte soit de x €, \$ (devise de l'instrument)
<code>TriangularAverage</code>	<code>TriangularAverage[N](price)</code>	Moyenne Mobile Triangulaire
<code>TRIX</code>	<code>TRIX[N](price)</code>	Triple Moyenne Mobile Exponentielle
<code>TypicalPrice</code>	<code>TypicalPrice[N]</code>	Prix Typique (moyenne de plus haut, plus bas et clôture)

U

CODE	IMPLÉMENTATION	FONCTION
<code>Undefined</code>	<code>a = Undefined</code>	Pour laisser une variable indéfinie
<code>UnSet</code>	<code>UnSet(\$MyArray)</code>	Réinitialise les données du tableau

V

CODE	IMPLÉMENTATION	FONCTION
<code>Variation</code>	<code>Variation(price)</code>	Différence entre la clôture de la veille et la clôture courante en %
<code>ViMinus</code>	<code>ViMinus[N]</code>	Bande inférieure de l'indicateur Vortex
<code>ViPlus</code>	<code>ViPlus[N]</code>	Bande supérieure de l'indicateur Vortex
<code>Volatility</code>	<code>Volatility[S, L]</code>	Désigne la volatilité de Chaikin
<code>Volume</code>	<code>Volume[N]</code>	Désigne le volume
<code>VolumeOscillator</code>	<code>VolumeOscillator[S,L]</code>	Désigne l'oscillateur de volume
<code>VolumeROC</code>	<code>VolumeROC[N]</code>	Désigne le volume du Rate Of Change

W

CODE	IMPLÉMENTATION	FONCTION
<code>Weeks</code>	<code>TIMEFRAME(X Weeks)</code>	Définit la période à « X Semaines » pour la suite des calculs du code.
<code>WeightedAverage</code>	<code>WeightedAverage[N](price)</code>	Désigne la Moyenne Mobile Pondérée
<code>WeightedClose</code>	<code>WeightedClose[N]</code>	Moyenne pondérée entre le prix de clôture, le plus haut et la plus bas
<code>WEND</code>	Voir <code>WHILE/DO/WEND</code>	Instruction à placer à la fin d'une boucle
<code>WHILE/DO/WEND</code>	<code>WHILE (condition) DO (action) WEND</code>	Boucle "Tant Que"
<code>WilderAverage</code>	<code>WilderAverage[N](price)</code>	Donne la moyenne mobile de Wilder

<code>Williams</code>	<code>Williams[N](close)</code>	Calcule le %R de Williams
<code>WilliamsAccumDistr</code>	<code>WilliamsAccumDistr(price)</code>	Indicateur Accumulation/Distribution de Williams

X

CODE	IMPLÉMENTATION	FONCTION
<code>XOR</code>	a XOR b	Opérateur logique OU exclusif

Y

CODE	IMPLÉMENTATION	FONCTION
<code>Year</code>	<code>Year[N]</code>	Donne l'évolution des années
<code>Years</code>	<code>TIMEFRAME(X Years)</code>	Définie la période à « X Année(s) » pour la suite des calculs du code.
<code>Yesterday</code>	<code>Yesterday[N]</code>	Donne l'évolution du jour d'avant

Z

CODE	IMPLÉMENTATION	FONCTION
<code>ZigZag</code>	<code>ZigZag[Zr](price)</code>	Zig-Zag de la théorie des vagues d'Eliott
<code>ZigZagPoint</code>	<code>ZigZagPoint[Zp](price)</code>	Zig-Zag calculé à Zp points

Autres

CODE	FONCTION	CODE	FONCTION
<code>+</code>	Opérateur d'addition	<code><></code>	Opérateur de différence
<code>-</code>	Opérateur de soustraction	<code><</code>	Opérateur d'infériorité strict
<code>*</code>	Opérateur de multiplication	<code>></code>	Opérateur de supériorité strict
<code>/</code>	Opérateur de division décimale	<code><=</code>	Opérateur d'infériorité
<code>=</code>	Opérateur d'égalité	<code>>=</code>	Opérateur de supériorité

ProRealTime SOFTWARE

